Pilot performance assessment using a hybrid expert system and machine learning for an automatic objective assessment in flight simulation

Jean-François Delisle
Polytechnique de Montréal
CAE Inc.
Montréal, Québec, Canada
j-f.delisle@polymtl.ca
jean-francois.delisle@cae.com

Andrea Lodi
Polytechnique de Montréal
Montréal, Québec, Canada
andrea.lodi@polymtl.ca

Maher Chaouachi,
Melvyn Tan,
Laurent Desmet
CAE Inc.
Montréal, Québec, Canada
maher.chaouachi@cae.com
melvyn.tan@cae.com
laurent.desmet@cae.com

ABSTRACT

An automatic pilot assessment capability using machine learning algorithms that can inform a flight instructor during a flight training session in full flight simulators is proposed in this paper. The current research explores a hybrid expert system and machine learning capability to assess pilot performance in flight simulation. Hybrid rule-based and machine learning algorithms are considered in the approach. Assessing a pilot's performance during a flight training session is a capability that can considerably improve the effectiveness of a training session and help the flight instructor provide better instructions and feedback. In this paper, we investigate an efficient way to build an automatic objective assessment engine, that provides a performance index that combines knowledge of subject matter experts and instructors' observations to train an artificial intelligence capability. By using multi-labels that have the same meaning but come from different sources of knowledge, we demonstrate that an automatic assessment engine is able to reduce the subjectivity of the instructor, optimize the time of the expert system's rule development effort. In addition, we show that this hybrid approach increases the accuracy and precision of the assessment of pilot maneuvers during training sessions by using a consensus methodology that blends the multiple sources of knowledge. The paper defines a data contextualization strategy using detection of training event windows on flight simulation data to classify pilot performance considering human bias, unbalanced datasets and variability of AI model maturity during an initial AI deployment.

ABOUT THE AUTHORS

Jean-François Delisle, CAE Inc., AI Innovation Lead, Advanced Technology and Innovation
Jean-François has over 25 years of experience in software engineering, AI, and data solutions. As a member of the Advanced Technology & Innovation department, he provides new solutions for the training and learning ecosystem. His research focuses on AI solutions that help optimize and adapt training delivery and the learning experience through human behavior and performance analysis. Jean-François joined CAE in 2010 and his current mandate is to define flight training technology strategies and disruptive innovation using data analysis and AI capabilities in Advanced Air Mobility and eVTOL engineering solutions. He earned a PhD in artificial intelligence and cognitive science for adaptive flight training under the supervision of Prof. Andrea Lodi, Canada Excellence Research Chair in Data Science for Real-Time Decision Making at Polytechnique Montréal.

Andrea Lodi, Andrea Lodi is the Andrew H. and Ann R. Tisch Professor at the Jacobs Technion-Cornell Institute, Cornell Tech and Technion–Israel Institute of Technology. His research interests include mixed-integer linear and nonlinear programming and data science. His work has been recognized by IBM and Google faculty awards and the 2021 Farkas Prize by the INFORMS Optimization Society. He has been Canada Excellence Research Chair in Data Science for Real-Time Decision Making at Polytechnique Montréal, network co-ordinator and principal investigator of EU and Canadian projects and consultant of the IBM CPLEX research and development team.

Laurent Desmet, Data Scientist, Digital Accelerator, CAE Inc. Laurent Desmet has a master's degree in data science from Polytechnique Montreal (M.Eng) and was hired by CAE three years ago. He started in the sound department where he developed an algorithm to harmonize acoustic energy, and then went to the Healthcare division of the company where he used his knowledge in several applications, from time-series forecasting to the analysis of images. Finally, he officially joined the data science team where he worked on a predictive engine to assess the performance of pilots during their training phase.

Maher Chaouachi AI Strategist, Digital Accelerator, CAE Inc. Dr. Maher Chaouachi holds a PhD in computer science specialized in the field of Artificial Intelligence obtained from the University of Montreal. He worked as a postdoctoral researcher at McGill University. Dr. Chaouachi is AI strategist at CAE working in various programs involving machine learning, process and data mining, optimization and predictive modelling.

Melvyn Tan, Data Analyst, Global Engineering, CAE Inc. Melvyn Tan joined CAE in 2019 and is part of the company's rotational development program. He has completed past stints at various sectors within aviation, including Scoot Airways in Singapore and Bombardier Aerospace in Canada. A believer in lifelong learning, he is currently completing his professional certificates in data science and artificial intelligence at the University of Toronto's School of Continuing Studies. He also holds a bachelor's degree in aerospace engineering from the University of Toronto.

Pilot performance assessment using a hybrid expert system and machine learning for an automatic objective assessment in flight simulation

Jean-François Delisle
Polytechnique de Montréal
CAE Inc.
Montréal, Québec, Canada
j-f.delisle@polymtl.ca
jean-francois.delisle@cae.com

Andrea Lodi
Polytechnique de Montréal
Montréal, Québec, Canada
andrea.lodi@polymtl.ca

Maher Chaouachi,
Melvyn Tan,
Laurent Desmet
CAE Inc.
Montréal, Québec, Canada
maher.chaouachi@cae.com
melvyn.tan@cae.com
laurent.desmet@cae.com

INTRODUCTION

With the ongoing evolution of flight training systems and their increasing complexity, it is necessary to have a robust approach to assist the instructor in the evaluation of pilot performance. Nevertheless, the evaluation of pilot performance by flight instructors has many drawbacks. This task can be labour-intensive and, furthermore, a human evaluator may not precisely evaluate all the flight parameters due to the limitations of human observational capabilities and the positioning of the instructor who is typically sitting behind the trainees. In addition, it is possible that the instructor has bias or simply uses a different evaluation standard compared to their peers.

In this current paper, we aim to provide a new system that will objectively assess pilot competencies in real-time and provide insightful and objective performance assessment of how pilots are performing during their training. An experiment was realized, during which data from several full flight training sessions was collected and processed to build a hybrid system that uses rule-based expert system and machine learning algorithms to automatically assess pilots' performance. The evaluation of the performance of computer-assisted pilots has been a subject of research for several years in the field of aviation. (Stein, 1984) examined a method to provide a performance index developed at the Federal Aviation Administration (FAA) Technical Center. As investigated by (Iqbal, Qadir, Mian, & Kamiran, 2017), grade-based Restricted Boltzmann Machines (RBM) technique prediction that aims to help students improve their performance and allow them to get the needed help from instructors. Collaborative filtering methods were used by (Rechkoski, Ajanovski, & Mihova, 2018) that presented an estimate of students' course grades to help them make decisions in order to achieve better results and obtain a degree in a timely and comprehensive manner.

Flight training involves the orchestration of training events in aircraft or flight simulators in accordance with a training curriculum. The flight training's objective is to provide pilots or crewmembers with an opportunity to acquire skills, attitudes, and knowledge of the standard operation procedure. The assessment is the measurement of a crew performance in the execution of a training event. Assessment criteria describe how the crew must perform the associated tasks. With (LeVie, 2016), the National Aeronautics and Space Administration (NASA) conducted a literature review to determine and identify quantitative standards for evaluating disruption recovery performance. This study contains current recovery procedures for military and commercial aviation and includes parameters for evaluating pilot performance in the context of upset prevention and recovery (UPRT) training in flight simulators.

Evaluating pilot performance is certainly accessible using data from actual aircraft. The authors of (Rantanen et al., 2007) describe measures of pilot performance that can be derived from data of the Flight Data Recorder (FDR). Standard deviation, root mean square error, number of deviations, time out of tolerance, and mean time to exceed tolerance are the used measurements. The work by (Stevens-Adams, Basilico, Abbott, Gieseler, & Forsythe, 2010) automatically evaluated student performance based on observed examples of good and poor performance in the assessment of tactical air engagement scenarios. The authors of (Chu, Gorinevsky, & Boyd, 2010) proposed an approach allowing precise detection of aircraft performance anomalies in cruise flight data. Detection is based on a model learnt from historical data of a fleet of aircrafts. Using the historical data, an average model is created based on the nominal vehicle operating data. No prior knowledge of the aircraft model is used, except knowledge of the inputs and outputs of the dynamic model. The flight dynamics model is empirically determined and identified from a set of flight data. Anomalies can be detected as deviations from the model. For a variety of cruising flight conditions with

and without turbulence, the authors validated the approach using a Flight Operations Quality Assurance (FOQA) dataset generated by a NASA flight simulator, where flight performance is monitored to ensure optimal operations and also to detect anomalies. They identified a regression model that maps flight conditions and aircraft control inputs. Anomalies are detected as outliers that exceed the dispersion caused by turbulence and modelling error. The detection method is related to the control of the multivariate statistical process. (Oza, Tumer, Tumer, & Huff, 2003) from the NASA Ames Research Center used in-flight data from two helicopters to propose a method where the offset between the actual flight maneuver in progress and the maneuver predicted by a classifier is a strong indicator of the presence of a fault. Nevertheless, to reduce high false alarm rates, it is important to understand the source of variability present in the flight environment. The authors of (Bryan Matthews, 2013) proposed a multivariate time-series search algorithm to search for anomaly patterns discovered in high-dimensional Commercial FOQA datasets. The process can identify operationally significant events due to environmental, mechanical, and human issues. The anomalies discovered were validated by a team of experts. The automated knowledge discovery process aimed to complement exceedance analysis done by humans, that fails to uncover previously unknown aviation safety incidents. In the paper of (Wang, Dong, Liu, & Zhang, 2015), the flight data history of thousands of aircraft flights are analyzed to provide a pattern recognition method based on the feature matching. This is adopted for automatic identification by analyzing the flight attitude of the aircraft and identifying the type of maneuver from the operational flight data.

Problem Statement

During a typical training session, the instructor is evaluating the pilot by direct observation. There is a substantial risk that they are not noticing all students' errors and missing the root cause of the pilot's actions and inaccuracies. In addition, multiple grading schemes exist in the industry. Grading a pilot performance is accomplished in a subjective manner and may be heavily influenced by instructor bias and cause of a non-standardization of global evaluations from one instructor to another. For a machine learning capability, using an instructor grade to train the system means using a subjective label and then fails the purpose of adding an automatic and objective assessment capability. There is a need to use a more reliable labeling strategy. Furthermore, training machine learning systems with instructor grading as the target label may be a concern for a new deployment of a new training program as we are confronted with a cold start issue. Therefore, there is a need for a deployment strategy that considers the progressive collection of labels as a source of truth. With a classic software engineering approach using a rule-based automatic assessment capability, a lot of pre-engineering efforts are required to implement all the various possibilities of performance. Rule creation and tuning cost are high when scaling for multiple aircraft and multiple training programs. There is a need to optimize the engineering phase with a more scalable technique using machine learning and transfer learning.

During a training session, only a few sequences are of interest to evaluate. The instructor needs to have an elevated level of attention to capture all events of interest and maintain his situation awareness all the time. With an automatic assessment capability, the problem is the same since the software will require to compute the entire training session data to perform the evaluation. There is a need to have a software capability that identifies the training session segment to be used for an evaluator, machine, or human. The algorithm must be able to identify the exact start and end time of the data segment of the flight sequence.

Objective & Hypothesis

Automatically assessing a pilot's performance during a flight simulator training session can enhance the performance of flight instructors and provide objectivity and interpretability during flight training assessment. We aim to explain the results of a pilot performance assessment by giving proper flight parameters that have importance into the decision-making process and how the model will behave in a production environment. By using multiple sources of truth, we are aiming to determine how we can integrate a few machine learning predictions together and apply consensus method, such as weighted majority voting to improve machine learning prediction accuracy and allow model management in cold-start context that corresponds to an AI system that has initially very low training data size.

In this paper, we will explore various machine learning techniques to assess pilot performance against a standard operation procedure. We will also create a Performance Index that standardizes the grading scheme and enabled the portability into multiple grading schemes. We will discuss the machine learning performance of an automatic grading capability against instructor grades and rule-based engine grades. We hypothesize that a combination of machine learning algorithms can increase accuracy by applying consensus functions such as weighted majority voting taking into consideration models maturity. This combination will increase performance accuracy and will provide better explainability of the machine learning model. With the proposed artificial intelligence solution architecture, we believe that we can reduce the flight engineer's workload to code all rules to detect and assess pilot maneuvers.

METHODOLOGY

A data collection phase was performed during flight training sessions executed by qualified professional pilots. The recording system saved more than 700 flight parameters (i.e., vertical speed, pitch angle, flap position, etc.) in the form of a time series with a frequency varying between 10 Hz and 60 Hz. Each training session is structured into training sequences called training events. More than 50 types of training events could be performed by the pilots during a training session such as crosswind landing, missed approach procedures, go around, abnormal and emergency procedures-reactive wind shear on take-off, and Low Visibility Operation (LVO) with Runway Visual Range (RVR) 150-400m. The selection of the training events to be performed was made by the instructor according to a predefined lesson plan. After each training event execution, the instructor grades the pilot's technical performance according to an ordinal 1 to 4 scale. It is defined as follows: Grade 1 – Fail, Grade 2 - Pass but room for improvement, Grade 3 - Meets expectations and Grade 4 - Exceeds expectations. During the training session, the instructors use an electronic grading system that automatically saves pilots' performance assessments on each performed training event. Each individual assessment is called a scorecard. From a three years data collection process, a total of 2,154 training event assessment scorecards were completed during 484 training sessions with an average of 4 different training events performed per training session. Overall, 210 different pilots, and 21 instructors were involved in this data collection.

Challenges related to the dataset

The main challenge in this research is that we are dealing with high multidimensional (over 700 features) time series classification problems. Each time series is a 2-minute sequence that corresponds to the performance of a single training event. Moreover, flight parameters that will be used in the artificial intelligence models are highly generally correlated (ex: altitude and airspeed are highly correlated with the landing maneuver). This has a consequence that requires multiple models' integration and is heavy on the explainability requirements. We have high multidimensional, both numerical and categorical features, ~700 collected, that leads to more complicated processing and that can be heavy on feature selection model too. Since the Flight Training industry is highly regulated, and our study population was certified commercial pilots, we had an Unbalanced Multiclass Classification problem. In an ordinal 1-4 scheme as the grading of the instructor, we had a lot of 2 and 3 representing the normal and pass behavior. Because 1 is a failure of the maneuver execution and 4 an exceptional performance rarely reached by pilots, we have very few 1 and 4 grades as presented in Figure 4. We are in the presence of Multiple Target Label with the same meaning but not in accordance, since one is coming from rules defined by a subject matter expert during the engineering phase of the cold-start approach, and the second is using grades by the instructor during official and regulated flight training sessions. Both rule-based and instructor grading are not guaranteed to be accurate and in accordance with each other. An expected difficulty is that the pattern to be detected may not have the same length as the set of datasets of the training set. Flight pattern recognition algorithms must be able to recognize patterns from different time ranges. This imposes large data collection to be able to recognize flight pattern profiles. That results in an important effort in data quality, data cleaning, and data pre-processing due to the overall size of the data. It also brings an important effort in real-time processing of time series data by preserving the quality of the flight simulation model that replicates at high frequency the flight model of an A320 aircraft manufactured by Airbus.

Solution Strategy

The general strategy is to use a hybrid system of expert and machine learning algorithms that can assess pilot performance from a flight training session data segment that corresponds to a flight training maneuver, as part of a flight training session. As presented in Figure 1, the data collection will begin with a local agent and a gateway already installed on a flight simulator that transfers data into a cloud platform data storage. The data is transformed into a Data Frame that is used in order to identify the capture window of a training event. Flight Phase and Training Event Detection algorithms are used to identify the segment of interest corresponding to a flight data time range. This will be used by the assessment module to compute a performance index based on key flight parameters involved in the maneuver. A flight training event sample is selected to associate the pilot maneuver data with a training context. This data will be evaluated (labelled) manually by an instructor, while automatically labelled by an expert system. This will be used to train machine learning models.

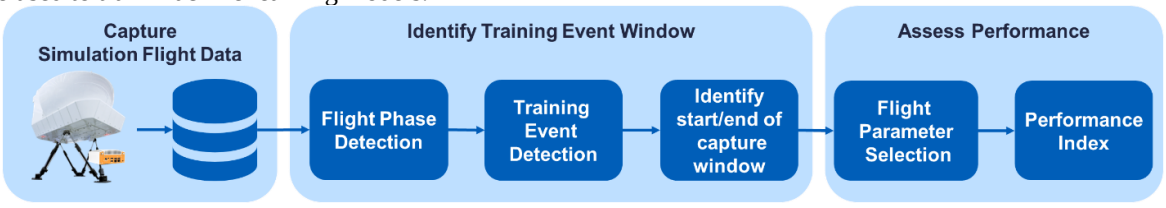


Figure 1 - End-to-end flow of a performance assessment on a training event

System Expert

The system expert methodology consists of creating training event rules and grading rules that are developed with subject-matter experts (SME) and interpreted by a software service. The rules are based on aircraft manufacturer standard operation procedures (SOP) and regulatory guidance to obtain the automatic performance assessment. Criteria and tolerances are commonly provided by the FAA and form the basis of instructor pilot evaluations. An existing expert system automatically detects training events and assesses pilot performance against a collection of standard maneuvers, in order to provide labelled data from flight SME knowledge. (Rule-based engine grading)

The Hybrid Machine Learning and System Expert Architecture

The machine learning models are deployed in an environment that has access to the flight simulator data during flight training in real-time. As presented in Figure 2, an electronic grading application that lists lesson plan steps and allows grading of a pilot maneuver is used by the instructor to grade pilots. The system expert is deployed and records rule-based grades from the same flight training data. Machine learning has advanced a lot in recent years, and is predominant with the advent of deep learning (Bengio, 2009). This method will be adopted in the first using multi-layer neural networks project and then tested in various architectures appropriate for time series data, such as Convolutional Neural Networks (CNN and Time Series Classification (Smith-Jentsch, Jentsch, Payne, & Salas).

Hybrid architectures are common in order to optimize solutions, add defined constraints, and respond to cold-start issues, such as (Cercone, An, & Chan, 1999) who presented a solution by combining Rule-Induction and Case-Based Reasoning in the machine learning framework. The researchers present a variety of machine learning techniques to improve the quality of the solution by combining rule induction methods with case-based reasoning techniques to improve performance compared to more traditional single-representation architectures. The author's article (Oladimeji, Turkey, Ghavami, & Dudley, 2015) presented us with a hybrid approach that involves the use of the k-means algorithm with neural networks during supervised machine learning that extracts patterns and detects hidden trends in complex data.

The following algorithms are used in parallel to optimize both accuracy and explainability depending on their respective advantages.

- *Support Vector Machine (SVM):* This method strives to find the boundary that best separates two different classes. It does so by identifying the extreme points of the dataset that are close to the opposite class, and a support vector is then drawn between these extreme points. A linear separator is then established between the two classes. Out of all the separation options, the model chooses the option that yields the largest distance between the two support vectors.
- *Deep Neural Network (DNN):* Designed after the structure of human brain neurons, they contain hidden layers that allow the model to identify different combinations of features to better carry out the required task (e.g., classification).
- *Convolutional Neural Network (CNN):* Like its name suggests, it contains convolutional layers that have filters that can identify patterns. Multiple layers can allow for identification of more complicated patterns/features. The key difference with the DNN approach is that convolution of operations are performed here. As such, the model input shapes for CNN would be different from that of DNN.
- *Decision Tree- Extreme Gradient Boosting (XGBoost):* It is a decision tree algorithm having multiple trees (bagging), randomly chosen features for each tree (random forest), using the output from one tree to the next for better performance (boosting), and using gradient descent to minimize errors (gradient boosting). Compared with gradient boosting, Extreme Gradient Boosting has multiple features built-in that enhance its performance. For example, it has a built-in cross-validation feature; to help the model to better learn from the data. L1 and L2 regularization are also used to prevent the model from overfitting. In terms of efficiency, this model is capable of parallel processing that helps decrease the time required to build it.

During the training process, hyperparameter tuning is conducted, as well as regularization (L1, L2 or Elastic Net), to prevent overfitting. We hypothesize that a feature selection module that learns which parameter has importance in the grading of a flight maneuver will contribute to scaling for multiple training events. The feature selection model will identify the important features involved in a specific task to scale the models for multiple maneuvers.

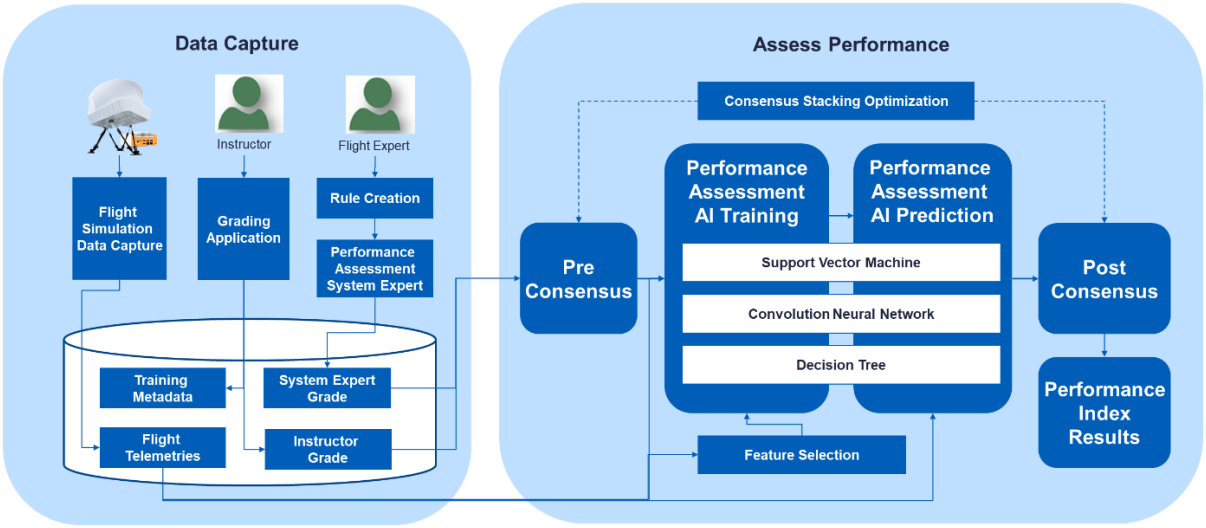


Figure 2 - Hybrid Machine Learning and System Expert Performance Assessment Architecture

Flight Phase and Training Event Detection

We identify time segments that correspond to training events and are the times of interest for pilot evaluation. By displaying them in the appropriate media, it allows instructors to save time in order to consult information for evaluation purposes or to demonstrate different concepts to students for pedagogical purposes in addition to feeding automated and real-time engines. Rule-based inference defined by a subject matter expert and analyzed by software services using a classic approach to detecting it is not always robust enough to fully capture all the possibilities and variations that define the context of an event. A particular difficulty lies in determining a good capture window, i.e. being able to predict with precision, within a time limit imposed by the rhythm of the training session, when an event begins and when it is finished. The uncertainty resides around the identification of each event for an increased accuracy rate even for a high maneuver complexity rate. We hypothesize that machine learning techniques can be used to increase detection capability using flight telemetry and some derived data.

Flight Phases divide a flight into separate sections with distinct characteristics, such as Climb, Cruise, or Descent. Flight phase identification is then an important marker to support training event detection. Recognizing flight phase and training event patterns automatically requires a lot of pre-engineering effort using a rule-based approach. Despite the important amount of data and flight engineering experience, recognizing the flight pattern requires a lot of authoring, tuning, and testing. This results in high cost, as well as low portability between aircraft types and between pattern types. Machine Learning Training Event Detection is proposed to complete the flow from the flight data segment to the end of the grading process. The more complex the maneuver, the harder it is to define the rules allowing the detection of training events by a classic rules-based software system. Moreover, a rules-based system is sometimes unable to detect certain events that have a high degree of complexity. We developed a machine learning model architecture to classify the presence of a training event in a training session evaluation. A Convolutional Neural Network (CNN) and XGBoost models were developed with instructor's evaluation metadata and time windows generated from the raw telemetry data. We trained and tested a machine learning model to identify the presence of a training event individually for each relevant training event.

Multi-Label Optimization Using Consensus as Ensemble Method

Since the quantity of data and their statistical property are variable, the maturity of the various machine learning models is also variable. By applying weighted consensus method on machine learning models according to their level of maturity; it forms a strategy where the combination of a variety of model's maturity is not leveling down the result. This strategy also allows combining multiple sources of knowledge, from flight training subject matter expert and certified flight instructor, as a human in the loop process that enhances the objectivity of the pilot performance assessment. We first used a Pre-Consensus method using the multiple target labels, each target with the same meaning (a pilot performance assessment grade), but not from the same source of knowledge (Rule-based objective assessment engine vs. instructor grades). With these two sources of labels, we applied the pre-consensus to obtain a new target label that will also be used in the training of the machine learning performance assessment models. We also trained with the perfect agreement set of assessment by the objective assessment rule-based engine and instructor grades. This

gives us either flawless performance or catastrophic outlier. We then combined with another data selection by removing exceptions and a separated set of perfect matches and worst matches in the assessment. After training, validation and test of the machine-learning algorithms, we apply a Post-Consensus methodology. It is the key strategy to manage the maturity of the models as a form of ensemble method that reduces the "Black box" side effect inherent to a deep learning model and can prevent providing good explanation for the flight instructor of the pilot performance. This is particularly useful during a cold-start approach where low maturity algorithms that might require different amounts of data can have less impact and degradation of the results compared to mature models. Various methods such as majority voting vs. weighted majority voting, where the weight is a maturity index composed of accuracy, lost, and F1-Score are used to optimize the results.

Dealing with unbalanced data

As shown previously, in terms of the grade labels, there is a lack of balanced data in the datasets. While there are tools from the Imbalanced Learn library, like over/under sampling and synthetic methods (e.g., SMOTE - Synthetic Minority Oversampling Technique), these did not yield very satisfactory results. Hence, there were a few different approaches that were combined to make up the strategy for treating the unbalanced data. We are using precision, recall & F1 score for the metrics, splitting the dataset for training to yield a less unbalanced dataset and using class weight to penalize over-represented classes. For hyperparameter tuning, we used Optuna (Akiba, Sano, Yanase, Ohta, & Koyama, 2019) a hyperparameter optimization software framework.

Using precision, recall, & F1 score for the metrics

As there is an unbalanced dataset, using precision and recall would be appropriate to evaluate the performance of the models before treatment for unbalanced data and after. As the F1 score incorporates both the precision and recall, this would be a convenient value to use for comparison. Using these metrics for the classification cases would be straightforward. However, for the regression case, the model's output would be a continuous grade value from 0 to 100%. To use precision, recall, and the F1 score, the model's continuous output would need to be reclassified back into the four discrete categories (1/2/3/4, or 25/50/75/100%). The way this was done is described in Table 1. The values found in the left column were obtained from the midpoints between each of the four discrete grade values. For example, the midpoint between 25% and 50% is 37.5%.

Table 1 - Continuous grade reclassification

Model's continuous output value	Categorized grade value
<37.5%	25%
>=37.5% & <62.5%	50%
>=62.5% & <87.5%	75%
>=87.5%	100%

Splitting the dataset for training to yield a less unbalanced dataset

To split data for training, considering the unbalanced dataset, the majority and minority classes in terms of data points are found from the training dataset. Now the amounts of data points in the majority classes are divided by the number of data points int the minority class, to yield the number of folds. Also, the training dataset is now split into two sections – one containing only the majority class data points ("majority dataset"), and the other containing all other points ("rest of datasets"). In each of the folds, the "rest of datasets" is combined with a fraction of the "majority dataset". This fraction is proportionate to the number of folds. For example, if the number of folds is 5, then one fifth of the "majority dataset" is joined to the "rest of datasets" to yield the new training dataset. The best dataset is then chosen for training, by using the metric of the average F1 score for all classes.

Using class weights and Optuna for hyperparameter tuning

Class weights are also used to address the class imbalance. To determine the size of class weights to use, the Optuna hyperparameter tuning framework is used. Float values between 0.1 and 20 are suggested for the four different grade classes. The average F1 scores across all classes are used as the metric for Optuna to optimize and obtain the maximum possible value. The best dataset from step 2) is used here to find the best class weights to use. If the dataset is being treated for instructor bias, the original instructor grade is used to assign class weights to that specific data point.

Instructor bias model

To remove this bias and standardize the scores, the average grade given by each instructor is calculated. Then, the average grade for all instructors is established. The average grade for each instructor is compared against the overall average for all instructors. This difference in value is measured against the overall average, to generate a percent difference. To calculate the adjusted grade, the original grade (1-4) is first multiplied by 25, to yield a maximum score of 100%. The percent difference is applied to this grade, so the final score is capped at 100%. This means that a student who receives a grade of 4 by a strict instructor should technically have a percent grade above 100%, but this cap prevents this from happening. Note that this treatment of bias is applied to the regression model, since its continuous nature allows for such tweaking of grades. Note that instructor bias treatment is only applicable for instructor grades, and only in the regression case. From the above equations, the grades will be adjusted by a certain percentage. As classification requires that the labels of data (grades) are discrete – 1, 2, 3, or 4, applying this grade adjustment would no longer yield discrete grade values. The following set of equations summarizes this approach.

$$avg\_all\_instructor\_grade = \frac{\sum avg\_instructor\_grade}{number\_instructors}$$

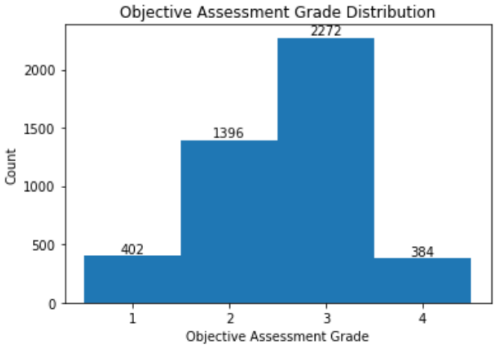
$$instructor\_diff = \frac{avg\_all\_instructor\_grade - avg\_instructor\_grade}{avg\_all\_instructor\_grade}$$

$$adjusted\ \%\ grade = \min\left[100, (1 + instructor\_diff) * original\_grade * 25\right]$$

Figure 3 - Instructor Bias Model

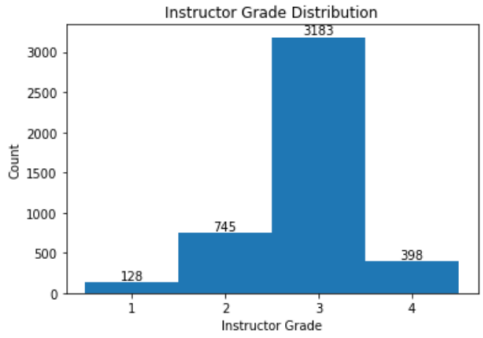
EXPERIMENTATION RESULTS

In this section we describe the obtained results of the experimentation of training event detection and performance prediction of the algorithms presented in the previous chapter. The analytics of this paper focused on the engine failure event at takeoff between the V1 and V2 takeoff speeds (EASA 2.5.2 GEN). We extracted this training event from the data generated between 2018 and 2020, and we were able to extract 4,454 instances of this event. These instances are conducted in 2,634 training sessions, on 22 training simulator devices. There are 1,921 pilots that have completed this particular training events, and 97 instructors that have graded pilots executed it. The Figure 4 and Figure 5 show the distribution of grades for both the objective assessment grade and instructor grade datasets. The objective assessment dataset has about half of the data points being a 3, with the grade of 2 as second most common.



Objective Assessment Grade Dataset	
Count	4454
Mean	2.59
Standard Deviation	0.77
Min	1
25%	2
50%	3
75%	3
Max	4

Figure 4 – Objective Assessment Rule-based system grade distribution & Statistics



Instructor Grade Dataset	
Count	4454
Mean	2.86
Standard Deviation	0.59
Min	1
25%	3
50%	3
75%	3
Max	4

Figure 5 - Instructor grade distribution & Statistics

To compare the objective assessment (rule-based) and instructor grades, the difference between these two grades is calculated. Out of the 4,454 cases, 3,132 of them are exactly the same (70% of cases). In these cases, the rule-based engine is accurate in predicting the instructor grade. To break this 70% down further by time of training events, the Table 2 illustrates this percentage by time period.

Table 2 - Rule-based vs instructor grade agreement per training cycle (6 months)

Period	Jul 2018 – Dec 2018	Jan 2019 – Jun 2019	Jul 2019 – Dec 2019	Jan 2020 – Jun 2020	Jul 2020 – Dec 2020
% of cases with no difference between rule-based and instructor grades	63%	60%	70%	71%	83%

Looking at the trends in the above table, with later time periods, there is an overall increase in the percentage of cases with the same grade in both the rule-based and instructor-based scenarios. In the event when the objective assessment does not predict the instructor grade accurately, 26% of the cases involve the situation where the instructor gave a higher grade than the rule-based engine. This means that the objective assessment is stricter in terms of evaluation. The remaining 4% of cases have the instructor grades being stricter. The distribution can be seen at Figure 6.

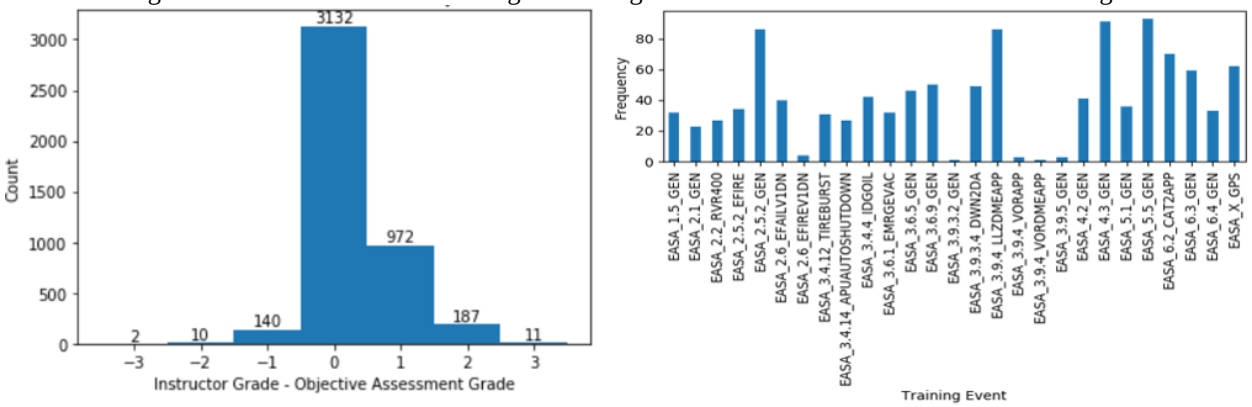


Figure 6 – Data Distribution - Difference between instructor and assessment engine & events detected using Rule-Based

Training Event Detection Results

A training event average length is 7 minutes for a maximum of 58 minutes. Figure 7 presents the distribution of detected training events using rule-based software services. It also presents the accuracy of rule-based event detection engine. The accuracy is based as a rate on the instructor's need to manually create the training event standardized by regulators and executed during a training session.

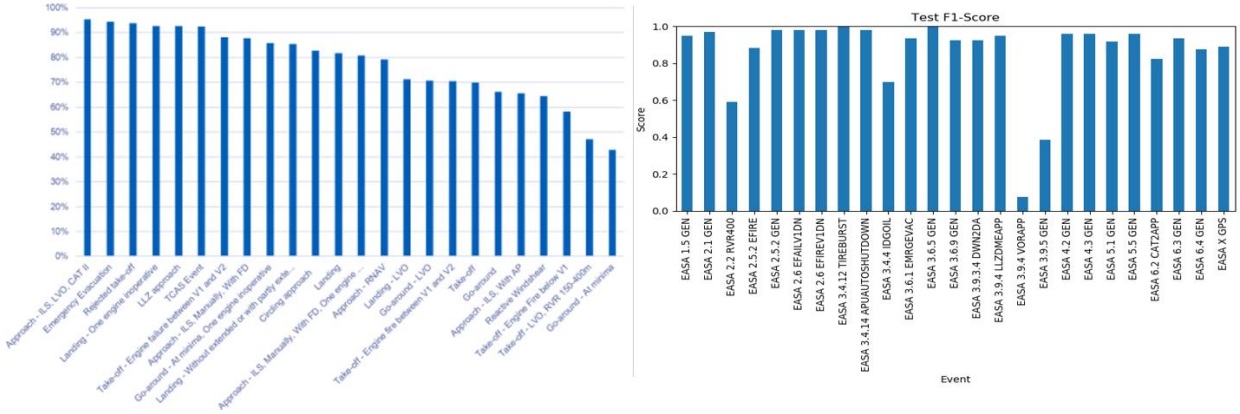


Figure 7 - Distribution of events detected using Rule-Based and Accuracy of Rule-Based event detection

As the different sensors that collect the data are set at different frequencies, the first step before feeding the data to the model is to change the data from the sensors to the same frequency. A neural network was initially chosen as the model type. However, the performance was not spectacular, and for that reason a tree-based network (XGBoost) was considered, to see how the two models compare. However, the XGBoost model requires that the sensor input be

changed to a tabular format. For that reason, the general statistics for each sensor (mean, standard deviation, max value, min value) is extracted, as well as the fundamental frequencies of the Fourier transform of each sensor. Lastly, with a limited number of samples, a data generator is also used to create more data samples. Except for a few outliers where the size of the dataset was insufficient as indicate in Figure 7, test F1-Score of over 90% of the detection rate was obtained using machine learning. At that rate, involving the contribution of an expert to analyze the 10% gap has become effective very much and can provide a great new characteristic to be included in a flight safety analysis and can be subject to review with aviation standards organization and regulators.

Figure 8 shows the results of training event detection using machine learning. Despite the addition of a machine learning model data generator, the results of CNN classification on telemetries as a time series recognition method still performed poorly given the small size of the dataset per maneuver type. The tree model with engineering features to identify training events as opposed to deep learning was seen as more appropriate. It was found that an XGBoost classifier with designed features was better suited. The tree-based algorithm used was gradient boosting in its implementation of XGBoost. Feature extraction was applied to the raw sensor data in order to represent the data fed into the model as a collection of feature values as opposed to a collection of sensors time series.

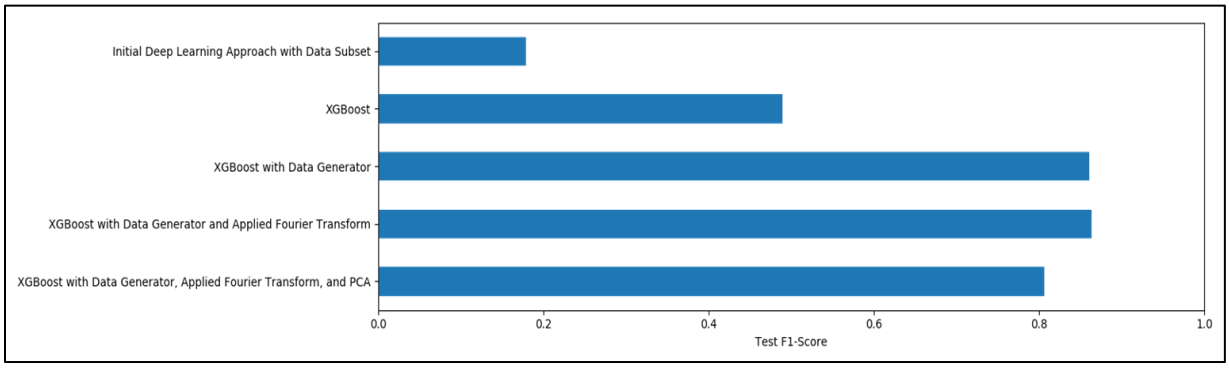


Figure 8 - F1-Score of Training Event Detection machine learning algorithms

Training Event Start-End Capture Window

We used a binary cross entropy for the detection and a Mean Absolute Error (MAE) for the regression on Window of 120 seconds. A deviation of 10 from the median of the Gaussian indicates a temporal standard deviation of 1 second. By analyzing the results of Figure 9, we can see almost 95% chance of having the correct prediction between +/-6 sec, 68% for +/- 3 sec. For an event that easily lasts 80 seconds, this makes a relative error of 5% compared to the duration of the event. However, with the low volume of data available per specific maneuvers, there is the potential of overfitting.

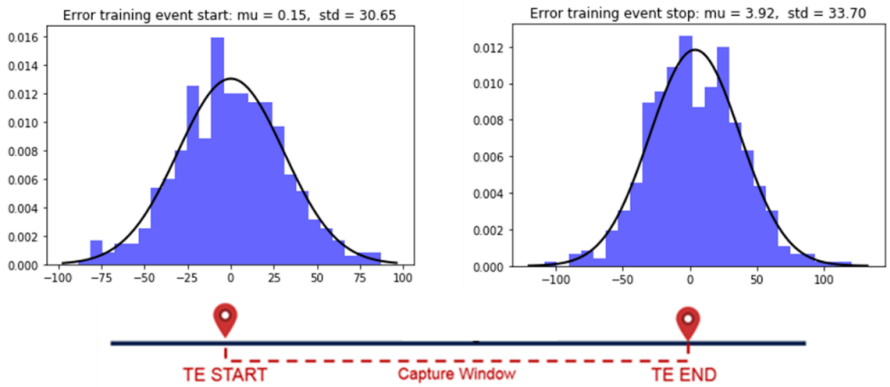


Figure 9 - Training Event Start and End time detection results to form the capture window

Performance Assessment Results

Figure 10 presents the accuracy of the machine models over data accumulation of pilot assessment data points. It presents the main SVM results for three supervised learning models that learned to use instructor grading, rule base grading as well as the consensus of both instructor and rule-based grading at 50% weight each.

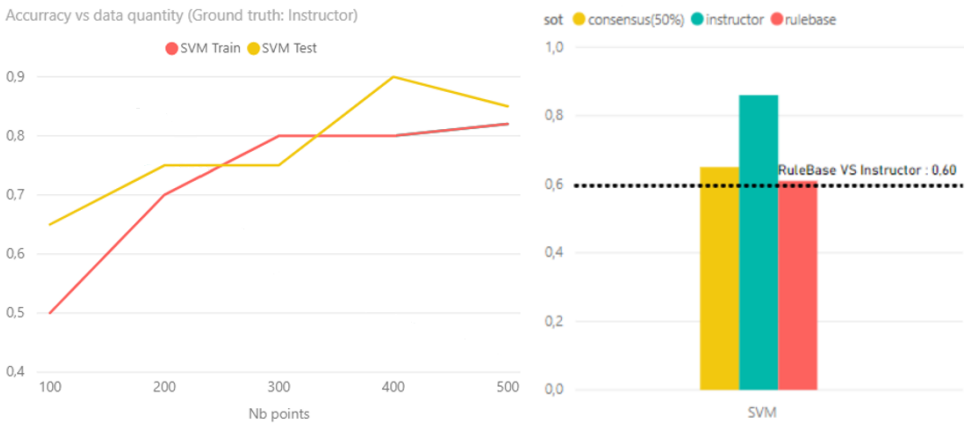


Figure 10 - SVM performance over data accumulation and compared to instructors and rule-based

We can see that machine learning SVM algorithm can grade technical performance like an instructor with the accuracy of ~85% vs. ~60% when using a rule-based assessment. We can determine how many maneuvers are required to reach the desired accuracy. We see that we can reach the 85% with around 500 data points using the machine-learning system by comparing them with instructor grade labels. It also demonstrates that the AI model can replicate the rule-based at 99% of accuracy, making this architecture a good candidate for a cold start approach at the deployment phase.

Dealing with unbalanced data

Our analysis comparing the baseline values with that of the balanced ones used the average F1 score over the four-grade classes. There is an increase in the average F1 score in 7 out of the 12 cases (4 different models x 3 grade types) For the 5 cases that did not show an improvement in the F1 score after balancing, 3 of them are related to the DNN model, while 2 of them are related to the SVM model. Nevertheless, it is important to know that the results above do not tell the whole story, since they do not reflect the distribution of F1 scores across the classes after balancing. Consider, for example, the Instructor grades with the DNN model as shows at the next table.

Table 3 - Baseline and Balanced DNN results

	DNN InstructorGrade bias_untreated Regression Baseline				DNN InstructorGrade bias_untreated Regression Balanced			
Results	25	50	75	100	25	50	75	100
Precision	0.814815	0.410714	0.755332	0.0909091	0.705882	0.240602	0.769841	0.131068
Recall	0.814815	0.150327	0.943574	0.0136986	0.888889	0.627451	0.304075	0.369863
F1	0.814815	0.220096	0.839024	0.0238095	0.786885	0.347826	0.435955	0.193548
Sample count	27	153	638	73	27	153	638	73

On the left are the baseline results, while the right shows the balanced results. Here, we see that there is a considerable improvement in the F1 score for both the 50 and 100 grade categories, at the expense of the 75 group. There is a negligible decrease in the 25 groups. One of the goals for balancing is to ensure the model pays more attention to the minority classes, instead of the majority of 75 group. This is done by splitting the dataset into two - the majority and the rest of the datasets and including a portion of the majority to the rest of the dataset for training. Therefore, while there is an overall decrease in the average F1 score across the grade categories, it is clear that the model is paying more attention to the other minority grade categories. This was the overall purpose of balancing the dataset.

Performance Index Calculation Using Regression on Multiple Labels and Pre-Consensus

To convert from a classification-based model to a regression-based one, there are a few modifications done. First, the label data is multiplied by 25. This converts the class-based data from 1-4 to a continuous one, from 0 to 100%. For XGBoost and SVM, the models are changed to use their regression instead of classification counterparts (XGBRegressor instead of XGBClassifier, SVR instead of SVC). For CNN, instead of having a softmax activation function in the last layer (as in the classification case), there is no activation function for the regression case. RELU was considered and may be used in later iterations. This allows the CNN model output to be continuous, instead of categorical. Comparing the performance in the classification and regression cases, out of the 12 cases (4 models x 3

grade types), there is a higher average F1 score in the regression case for 9 out of the 12 cases. For the remaining 3 cases where the regression case yields a lower F1 score than the classification case, all of them are related to the DNN model. A possible reason for a higher F1 score (and better performance) in the regression case is that there is more flexibility for the model to be more accurate when quantifying the performance of the pilot. Nevertheless, when making this conclusion, one needs to acknowledge that there is also an extra step of reclassifying the continuous grades into discrete values in the regression case.

Consensus and model management

Our analysis provided post consensus results of the machine learning regression models for instructors, rule-based and pre-consensus target labels. Weighted mean contains the following weights for the different models: CNN – 0.4, DNN – 0.1, XGB – 0.25, SVM – 0.25. Using the instructor grades for the model input would yield a better prediction of the instructor grades, as given by the lower RMSE values. For the three different grades (OA, Instructor, and Consensus), the CNN model requires a minimum of 300 samples, before achieving a low RMSE value that is comparable with that of the other models. This minimum size is not observed with the SVM or XGBoost models; both offer a low and relatively steady RMSE from a sample size of 100.

Instructor bias results

To apply the bias treatment, the bias treated instructor grades are given to the model as inputs. This model will then balance the data as described in the unbalanced data section. In order to evaluate the performance of the models, the RMSE values for the balanced cases and the bias treated balanced cases are compared. Instead of using precision and recall as a metric like in the balanced treatment section, the RMSE is used. The reason for this is that the bias treatment adjusts the instructor grade by a certain percentage. This percentage might not yield a distinct difference, when we reclassify the results back into the four grade categories to calculate the precision and recall. However, the RMSE should allow us to identify if there is a clear improvement in the model's performance after treating for instructor bias. A lower RMSE would mean a more accurate model. From our analysis results, there is an improvement in the RMSE score from all four model types, when the instructor bias treatment is applied to the balanced dataset.

CONCLUSION

This research was developed to help instructors deliver training in accordance with airline standard operating procedures (SOPs). Key benefits of the system include the ability to objectively assess pilot skills in real time and provide insightful training analytics in the eventual support of a Threats and Error Management (TEM) and competency-based frameworks for Evidence-Based Training (EBT) support that shall be the next shift in pilot performance assessment. Rule-based system is specifically built for aircraft models and maneuvers, and not easily transferable to new instances of the cases. An automatic rule-based assessment engine cannot scale and cover every possible variation that could impact a human expert's assessment of student performance. The typical rule-based inference to detect it is not always robust enough to fully capture all the possibilities and variations that a flight maneuver might be subject to. The uncertainty here lies in how well we can fully assess performance by appreciating the simulated conditions and context. With artificial intelligence, we can make the data speak at a level unattainable by rules in an effective way. The machine learning strategy presented in this paper was successfully integrated to rule-based to provide a hybrid solution that can support machine learning models training and cold-start deployment considering multiple sources of knowledge. The usage of a form of ensemble method demonstrates that the explainability of the models as well as an enhance model management method can provide a valid solution for a robust automatic flight performance assessment capability.

Reinforcement Learning is another machine learning technic that can be explored. With an instructor in the loop that can provide feedback to the engine, it can improve the grading in runtime in a Man-Machine Teaming interaction in a distributed situation awareness context of the pilot performance. Semi-supervised learning can also be an option where consensus maximization with parameter estimation can be applied to consider that data may be missing (missing instructor grades for a few training events). Transfer learning can be applied to be able to reuse learning models to more than one task to be automatically assessed. Transfer Learning techniques can be used to generalize the models for multiple maneuvers and with a limited size of data sample. Federated machine learning has horizontal, vertical, and transfer learning capability that can certainly be used in future work in order to combine multiple aircraft models and enable the scalability and cost reduction for the deployment on multiple training programs from multiple training centre. The approach can also be used for Machine Learning Model Transferability to multiple maneuvers that provide a strong difference in the flight profile (Straight Approach, Landing, Approach with go-around, etc.). This will address scalability for multiple aircraft and multiple training curriculums. The Gap Analysis Model will learn the difference

between the instructor and the rule-based grading to scale multiple aircraft where the rule-based engine will not be present and where the instructor grading and the Gap Analysis Model will be used together to obtain similar results from an aircraft that will have both grades. The system generated insights intended to help the instructor understand the performance of the pilots. A key element of performance evaluation is the identification of the root causes that led to flight exceedances. The root cause may not be explicitly captured in the simulated raw data. For example, a lack of a pilot's situational awareness or decision-making can lead to a late reaction time or wrong course of action. The use of the neuroscience can then take on all its interest. A model for integrating technical skills and non-technical skills in assessing pilots' performance can be used as mentioned by (Mavin 2010). Standardization of the grading can also be done using biometrics of the instructor to capture situation awareness during the evaluation of the pilot. Finally, this research focused on the assessment of the pilot's technical skills that have specified performance parameters. To cover a full assessment, we should increase soft skills as well as crew resource management (CRM) to support Advanced Qualification Program (AQP) and Line Oriented Flight Training (LOFT)

ACKNOWLEDGEMENTS

Mehdi Taobane, Project Manager, DS4DM, Polytechnique de Montreal, Marc St-Hilaire, CTO, CAE Inc.; Mark Soodeen, AT&I Director, CAE Inc, Houssam Alaouie, AT&I Director, CAE Inc.; Patricia Gilbert, Project Manager, CAE Inc.; Anthoine Dufour, Data Scientist, CAE Inc., Yang Meng, Student, Trent University, Dac Toan Ho, Software Engineer, CAE Inc., Marc-André Proulx, Software Architect, CAE Inc., Samir Sahli, AI Advisor Microsoft, Jean-Mathieu Deschênes, Data Engineer, CAE Inc., Alejandro Sanchez, Data Scientist, Lixar Inc.

REFERENCES

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). *Optuna: A Next-generation Hyperparameter Optimization Framework*. Paper presented at the Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.

Bengio, Y. (2009). Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*.

Bryan Matthews, S. D., Kanishka Bhaduri, Kamalika Das, Rodney Martin, Nikunj Oza, Ashok N. Srivastava, John Stutz. (2013). Discovering Anomalous Aviation Safety Events using Scalable Data Mining Algorithms. *Journal of Aerospace Information Systems, 10*(10).

Cercone, N., An, A., & Chan, C. (1999). Rule-Induction and Case-Based Reasoning Hybrid Architectures Appear Advantageous. *IEEE Transactions on Knowledge and Data Engineering, 11*(1).

Chu, E., Gorinevsky, D., & Boyd, S. P. (2010). Detecting Aircraft Performance Anomalies from Cruise Flight Data. *Proceedings AIAA Infotech@Aerospace*.

Iqbal, Z., Qadir, J., Mian, A. N., & Kamiran, F. (2017). Machine Learning Based Student Grade Prediction (A Case Study). *arXiv*(arXiv:1708.08744).

LeVie, L. R. (2016). *Survey of Quantitative Research Metrics to Assess Pilot Performance in Upset Recovery*. Retrieved from

Oladimeji, M. O., Turkey, M., Ghavami, M., & Dudley, S. (2015). *A New Approach for Event Detection using k-means Clustering and Neural Networks*. Paper presented at the International Joint Conference on Neural Networks (IJCNN).

Oza, N. C., Tumer, K., Tumer, I. Y., & Huff, E. M. (2003). *Classification of Aircraft Maneuvers for Fault Detection*. Paper presented at the International Workshop on Multiple Classifier Systems.

Rantanen, E. M., Talleur, D. A., Taylor, H. L., Bradshaw, G. L., Tom W. Emanuel, J., Lendrum, L., & Hulin, C. L. (2007). Derivation of pilot performance measures from flight data recorder information. *IEEE Aerospace and Electronic Systems Magazine*.

Rechkoski, L., Ajanovski, V. V., & Mihova, M. (2018). *Evaluation of Grade Prediction using Model-Based Collaborative Filtering methods*. Paper presented at the IEEE Global Engineering Education Conference (EDUCON).

Smith-Jentsch, K. A., Jentsch, F. G., Payne, S. C., & Salas, E. (1996). Can pretraining experiences explain individual differences in learning? *Journal of Applied Psychology, 81*(1), 110-116. doi:10.1037/0021-9010.81.1.110

Stein, E. S. (1984). *The Measurement of Pilot Performance. A Master Journeyman Approach*. Retrieved from

Stevens-Adams, S. M., Basilico, J. D., Abbott, R. G., Gieseler, C. J., & Forsythe, C. (2010). *Performance Assessment to Enhance Training Effectiveness*. Paper presented at the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC).

Wang, Y., Dong, J., Liu, X., & Zhang, L. (2015). Identification and standardization of maneuvers based upon operational flight data. *Chinese Journal of Aeronautics, 28*(1), 133-140. doi:10.1016/j.cja.2014.12.026