Advancing Aircraft Operations in a Net-Centric Environment with the Incorporation of Increasingly Autonomous Systems and Human Teaming

Vincent E. Houston[1]
NASA Langley Research Center, Hampton, VA, 23681

Walter J. Manuel[2]
Purdue University, West Lafayette, IN 47907

Evana Gizzi[3]
Tufts University, Medford, MA 02155

Bryan A. Barrows[4]
NASA Langley Research Center, Hampton, VA 23681

NextGen has begun the modernization of the nation's air transportation system, with goals to improve system safety, increase operation efficiency and capacity, provide enhanced predictability, resilience and robustness. With these improvements, NextGen is poised to handle significant increases in air traffic operations, more than twice the number recorded in 2016, by 2025.[1] NextGen is evolving toward collaborative decision-making across many agents, including automation, by use of a Net-Centric architecture, which in itself creates a very complex environment in which the navigation and operation of aircraft are to take place. An intricate environment such as this, coupled with the expected upsurge of air traffic operations generates concern respecting the ability of the human-agent to both fly and manage aircraft within. Therefore, it is both necessary and practical to begin the process of increasingly autonomous systems within the cockpit that will act independently to assist the human-agent achieve the overall goal of NextGen. However, the straightforward technological development and implementation of intelligent machines into the cockpit is only part of what is necessary to maintain, at minimum, or improve human-agent functionality, as desired, while operating in NextGen. The full integration of Increasingly Autonomous Systems (IAS) within the cockpit can only be accomplished when the IAS works in concert with the human, formulating trust between the two, thereby establishing a team atmosphere. Imperative to cockpit implementation is ensuring the proper performance of the IAS by the development team and the human-agent with which it will be paired when given a specific piloting, navigation, or observational task. Described in this paper are the steps taken, at NASA Langley Research Center, during the second and third phases of the development of an IAS, the Traffic Data Manager (TDM), its verification and validation by human-agents, and the foundational development of Human Autonomy Teaming (HAT) between the two.

Nomenclature

ASTRAO	=	Autonomous System Technologies for Resilient Airspace Operations
D.A.T.A	=	Dynamic Air Traffic Application
HAT	=	Human-Autonomy Teaming
IAS	=	Increasingly Autonomous Systems

[1] Computer Research Engineer, Crew Systems Aviation and Operations, NASA Langley, Senior Member
[2] Undergraduate Aerospace Engineering Student, Purdue University
[3] PhD Candidate, Artificial Intelligence, Tufts University
[4] Mechanical Engineer, Autonomous Integrated Systems Research Branch, NASA Langley

MATIMAL	=	Machine Learning Algorithms To Improve Model Accuracy and Latency
ML	=	Machine Learning
MVEL	=	Multi-View Ensemble Learning
ND	=	Navigation Display
NextGen	=	Next Generation Air Transportation System
NAS	=	National Airspace System
RAS	=	Relative Altitude Squared
SASO	=	Safe Autonomous Systems Operations
SVM	=	Support Vector Machine
TDM-ML	=	Traffic Data Manager-Machine Learning
TDM-PM	=	Traffic Data Manger-Prediction Model
VISTAS	=	Visual Imaging Simulation for Transport Aircraft Systems
V&V	=	Verification and Validation

I. Introduction

Since 2009, a 3.8 percent annual increase in air traffic operations within the NAS has been observed. Maintaining the current rate of increase translates to ≈ 96,000 operations taking place daily, in the United Stated, by 2025.[1] The challenge represented by this statistic begs the question: How can flight safety and operational integrity be maintained considering the aforementioned rise and pace of aircraft operations in the NAS? The consistent increase of air traffic operations requires a NextGen net-centric environment in order to handle this future projection. A Net-Centric environment offers many advantages and, of particular interest, is the provision for collaborative decision-making using distributed automation among hundreds of agents. However, a complex community, such as NextGen, consisting of personnel, devices, information, and services does have its hurdles, which must be overcome so that those benefits can be unlocked.

NASA is investigating the development of machine-agent technology that would enable Net-Centric operations within this type of densely populated airspace and complex community. Our work consists of the development of Increasingly Autonomous Systems (IAS). IAS are envisioned as intelligent machines - hardware and software systems - seamlessly integrated with humans, whereby task performance of the combined system is significantly greater than the individual components.

The formulation and implementation of IAS within the cockpit leverages the human-agent's flight experiences and expertise with the computational robustness, processing speed, information storage, distraction immunity, and learning ability of the machine-agent in order to build a symbiotic relationship between human and machine agent. Functionally, autonomous agents, human or machine, act independently within a delineated set of criterion; then either take action or delegate responsibility to another agent in order to achieve the overall system objective(s). IAS maturates current automated systems by having a greater level of intelligence built into its system architecture. Utilizing adaptive learning capabilities that allow it to respond within a circumscribed set of goals in response to situations that were not hard-coded or anticipated in the design. This concept is fundamental to an instantiation of 'intelligent party-line', where the human-agent is properly apprised of his/her flight environment, without suffering information-type overload, during computer-to-computer interactions within a NextGen Autonomic Airspace Architecture.[5]

An initial technological development in the area of IAS was conducted during Phase One of Autonomous System Technologies for Resilient Airspace Operations (ASTRAO).[2] The focus of ASTRAO (Phase One) research was to utilize a data-lake comprised of eight different learning features and ≈22K data points for training multiple supervised Machine Learning (ML) algorithms in determining aircraft relevance with respect to an ownship for a human's awareness of traffic in a Net-Centric environment. Further, cross-validation was performed on the ML algorithms in order to assess the ability of the model to predict new data and to provide insight on how the model will generalize to an independent dataset. During Phase One, MATLAB's TreeBagger ensemble method proved the better performer in predicting "not-relevant", "maybe-relevant", and "relevant" air traffic with accuracies of 88.7%, 48.5%, and 46.7% respectively, and an overall accuracy of 61.3%.[2] The outcomes from ASTRAO were skewed heavily toward non-relevant air traffic. This was the result of the SMEs' relevancy determinations biasing the TDM prediction model toward "not-relevant" during training. As a consequence of this, the approach to Phases II and III of ASTRAO (Machine Learning Algorithms To Improve Model Accuracy and Latency (MATIMAL)) was twofold: 1) develop a more sophisticated and robust ML algorithm that would offer greater accuracy in the prediction of relevant air traffic with respect to an ownship; and, 2) extrapolate this learning process and algorithmic schema toward the predictions of "maybe" and "not-relevant" air traffic with respect to an ownship. In addition, apply algorithm feature enhancements to the newly developed ML algorithm as well as implement a more complimentary and robust way of validating and verifying the prediction accuracy of TDM while fostering Human-Autonomy Teaming (HAT).

II. Study Design

The Traffic Data Manager (TDM) is a multi-subsystem software application, with an embedded autonomous element, designed to acquire air traffic surveillance data, with respect to an ownship, along with utilizing Subject Matter Expert (SME)-captured knowledge to learn patterns of air traffic relevancy in order to partition individual aircraft in a traffic pattern into categories of Not-Relevant, Maybe-Relevant, and Relevant air traffic.[2]

This paper details the progression of TDM's prediction model (TDM-PM) from its prototype phase during ASTRAO to its developmental phases of MATIMAL - the algorithmic augmentations implemented to improve the model's relevant inferential accuracy. This research continues to build foundational basis critical to NASA's development of IAS to achieve an enabling technology for Net-Centric NAS operations. The objective, in this case, was to capture and leverage a pilot's expert knowledge of air traffic distinctions, then utilize this state and contextual information that defines this knowledge, to teach a machine to make air traffic relevancy predictions. The work completed during MATIMAL II & III marks the end of the prototyping phase of TDM-PM and signifies the beginning of human-agent verification and validation of machine-agent inferences.

A. MATIMAL Phase II (Design Structure, Training)

A machine-agent acquires intelligence through learning what a human-agent knows in accordance with a particular task, this is accomplished through the implementation of the ML workflow as shown in Figure 1. The Traffic Data Manager (TDM) is a multi-system IAS application developed to establish intelligent party-line capabilities and improve flight crew situation awareness through the partitioning of air traffic. Utilizing the surrounding aircrafts' state data TDM learns patterns of air traffic relevancy and then segments individual aircraft, within that traffic pattern, into categories of Not-Relevant, Maybe-Relevant, and Relevant aircraft. TDM's design structure during MATIMAL had five different subsystems that made up its core.[6]

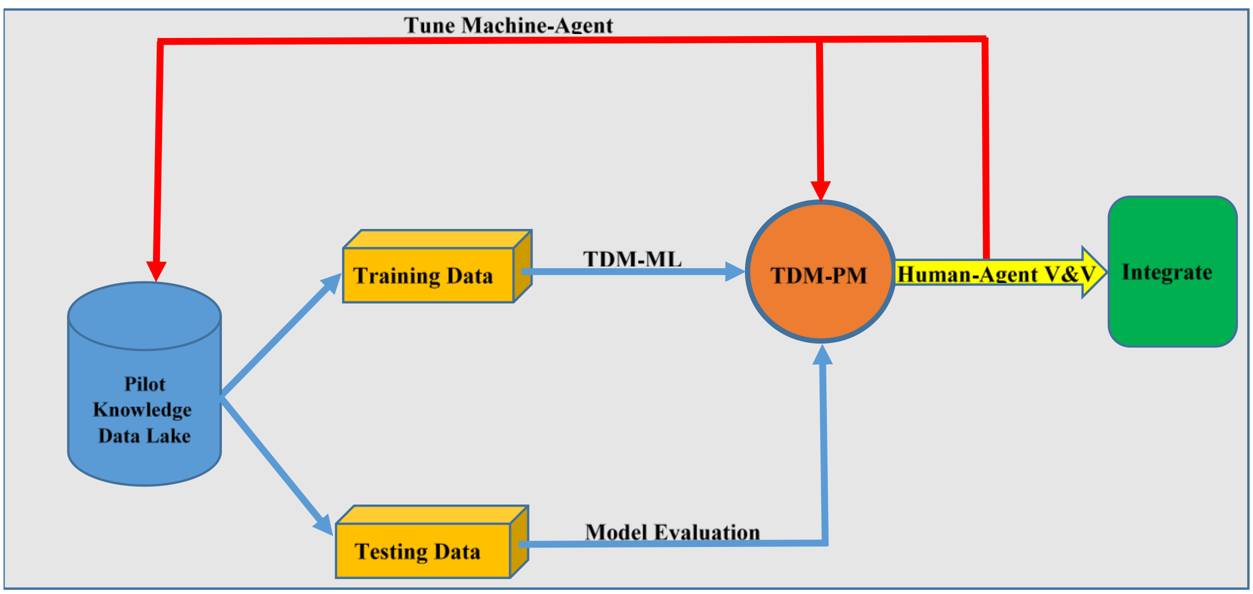

Figure 1. TDM Supervised Machine Learning Workflow

The following five subsystem elements are essential to TDM's development:

- Flight Aware Data Grabber (FLTAWR-DG) –a MATLAB-developed software application that performs a data scrape from the "Flightware-Live" website and records the surveilled air traffic state data. Executing file formatting procedures, it then records subject matter expert (SME, human-agent)-specified state data labels into MS Excel™ files for training the TDM-PM. Figure 2 illustrates the end result of the data grab process carried out by FLTAWR-DG.

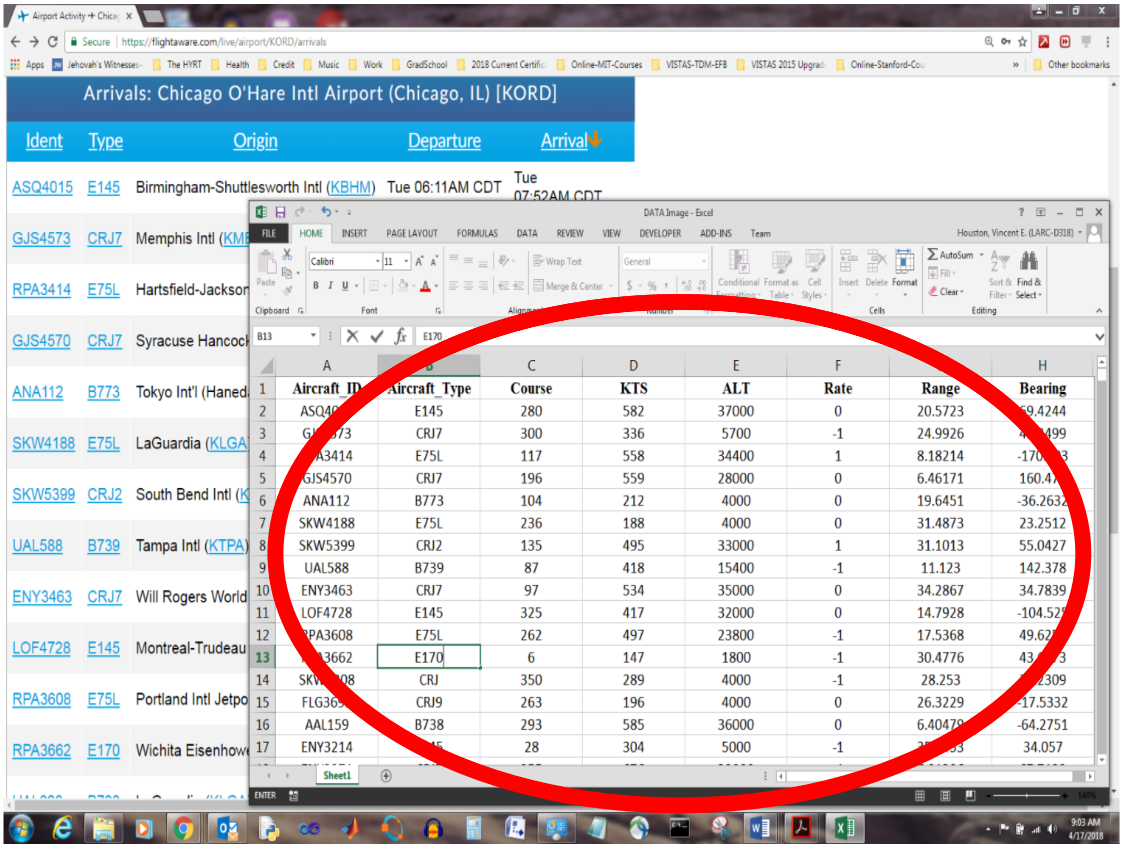

Figure 2. Flight Aware Data Grabber (FLTAWR-DG)

- Dynamic Air Traffic Application (D.A.T.A) –TDM's interface application and was created to render air traffic scenarios on a Navigation Display (ND) for data collection and algorithm verification and validation (Figure 3). D.A.T.A, created a data lake of air traffic relevancy determinations by allowing SMEs to record their knowledge of "Relevant", "Maybe-Relevant", and "Not-Relevant" aircraft, with respect to an ownship. TDM-PM then acquired its aircraft relevancy intelligence by training TDM's ML algorithm from that data lake. In addition, D.A.T.A rendered similar scenarios of air traffic patterns generated by relevance predictions on new data provided to TDM-PM for the verification and validation, by human-agents, of TDM-PM. In the case of either training or verification and validation (V&V), "relevant" always meant those aircraft that a pilot deemed that they needed awareness for the safe and efficient operation of their flight.

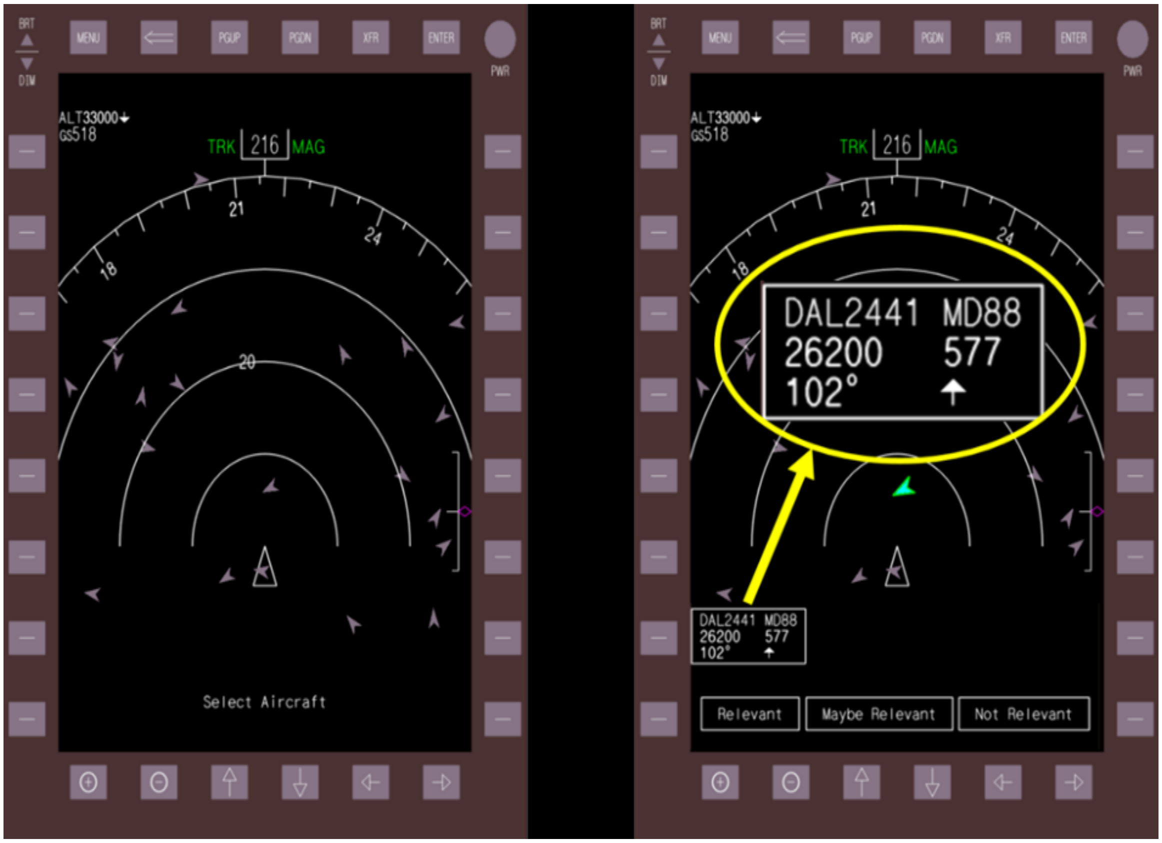

Figure 3. Dynamic Air Traffic Application (D.A.T.A)

- Traffic Data Manager-Machine Learning (TDM-ML) – An embedded ML algorithm trained to correctly classify air traffic autonomously. TDM-ML utilizes aircraft state data/features (course, airspeed, altitude, rate, range, and bearing) from air traffic scenarios after training with correct examples of aircraft relevance provided by the SME with respect to an ownship, in order to correctly infer the relevance of future aircraft with respect to ownship. TDM-ML is also used to provide continuous training on new data, if it is desired.
- Traffic Data Manager-Prediction Model (TDM-PM) – An artifact created by the training process of TDM-ML which is embedded in the machine learning process. TDM-PM captures the patterns that TDM-ML has learned to identify from the correct answers (target attributes) provided by the human-agent. It can then generate predictions on new data when the target attribute or correct answer is not known.
- Navigation Display (ND) – The display of traffic aircraft positions with respect to ownship within an air traffic scenario. Aircraft state data captured by FLTAWR-DG was used to initially populate the ND with an unfiltered view of aircraft traffic with respect to the ownship. Once TDM has been trained and its model renders aircraft relevancy predictions, the ND then repopulates with air traffic scenarios according to their inferred relevance of *"Not-Relevant"*, *"Maybe-Relevant"*, and *"Relevant"*.

The lessons-learned in Phase I prototype development of TDM caused the formulation of a different instantiation of supervised ensemble learning being applied during development Phases II & III (MATIMAL). The improved ML algorithm implemented during MATIMAL was Multi-View Ensemble Learning (MVEL). The following sections of this paper will address the details of the MVEL algorithm, the enhanced features implemented, human-agent V&V, and data analysis.

B. Multi-View Ensemble Learning (MVEL)

MVEL is an approach to data analysis and machine learning where a complete data set is segmented into multiple distinct feature subsets. These subsets incorporate various predictors, from the aforementioned data set, toward target concept learning for a standalone supervised classifiers. TDM-ML, and subsequently TDM-PM, consist of four feature subsets and four different classifying algorithms. The first of the four algorithms was the Bootstrap-Aggregated Decision Trees (TreeBagger), which was used previously in ASTRAO and is governed by its prediction equation.[2] Another ML classifier used in MATIMAL was K-Nearest Neighbor (K-NN). K-NN is a non-parametric classification method used to assign weightings to the contribution of features taken from the data subset. K-NN utilizes the following posterior probability function $p(j|X_{new})$ to recognize patterns in the feature subset and infer air traffic relevance with respect to ownship. The truth of this statistical probability is derived from $p(j|X_{new}) = \frac{\sum_{i \in nbd} W(i)\, 1_{Y(X(i)=j)}}{\sum_{i \in nbd} W(i)}$ representation. Naïve Bayes and Support Vector Machine (SVM) round out the four algorithms used to create the ensemble feature of MVEL used in MATIMAL.

The Naïve Bayes classifier is a simple "probabilistic classifier" based on the application of Bayes' theorem with significant (naïve) independence assumptions between the features. The Naïve Bayes classifier combined its independent decision model with a common decision rule to select the hypothesis that is most likely to occur; known as the maximum posteriori (MAP) decision rule. The resulting classifier is a function that assigns a prediction $\hat{y} = C_k$ for some k such that: $\hat{y} = \underset{k \in \{1,\ldots,K\}}{argmax}\, p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$.

The final classifier in the MVEL ensemble is the SVM. The SVM classifier inputs training examples from the data set, marking each as belonging to either one of the three air traffic relevance categories (relevant, maybe-relevant, not-relevant). The model built from training the SVM algorithm assigned new examples to one category of relevance or the other. This represents a non-probabilistic binary linear classifier of the examples as points in space, which were mapped in order for the data points of separate air traffic relevancy categories to be divided by a gap that is as wide as possible. New input data points were then mapped into that same space and predicted to belong to a particular relevance category (relevant, maybe-relevant, not-relevant) based on which side of the space they resided.

The algorithmic computation for training the SVM model used in MATIMAL was defined by the following posterior probability function: $P(s_j) = \frac{1}{1+e^{\left(As_j + B\right)}}$, where A and B are the slope and intercept of the predictors. [7, 10, 11] The following equation provided the new vector observation weights for the model inferences based on the largest posterior probability ($W_j = w_j P_j \sum_{k=1}^{K} C_{jk}$), were *P* represents posterior vector, *C* is the misclassification matrix, *w* is the observation weight vector.

The collaborative nature of MVEL required that optimization of the outcome related to the target concept need be incorporated. Plurality voting, to ensemble the individual models was carried out. MVEL's ability to exploit multiple views of the input data generated a superior learning and pattern interpreting functionality for TDM's ML. With the implementation of MVEL, TDM was better able to predict relevant air traffic with respect to an ownship then it was using only the TreeBagger algorithm during ASTRAO.

C. MATIMAL Enhanced Feature Selection

Training the TDM-ML algorithm, during ASTRAO, to make plausible inferences as to the relevance of air traffic, required aircraft type, course, airspeed, altitude, vertical trend, range, and bearing data along with SME responses for predictor (feature) input to the algorithm. However, after the execution of ASTRAO, consideration was given to the significance of those features and whether their use toward training TDM-ML could be improved by either the addition of different features or by narrowing the scope of the features to only the most relevant ones.

Phase II of the TDM development initializes training of the MATIMAL MVEL algorithm with the complete data set of predictors from ASTRAO and the corresponding SME responses. Feature analysis and relational data tests were performed on those existing features in order to determine which features offered the most promise in improving the accuracy of relevant air traffic predictions inferred by TDM-PM. Applying unsupervised learning algorithms K-Means clustering, Principal Component Analysis (PCA), and t-distributed Stochastic Neighbor Embedding (t-SNE) provided a better understanding of the dataset and revealed hidden relationships between features. However, the analysis demonstrated that, although the requisite features proved beneficial in that they fully captured the complete state of the surrounding air traffic, it was not enough to overcome the inconsistent prediction of air traffic relevance with respect to ownship. Therefore, it was determined that derivations of the features would be necessary to ensure that the traffic aircraft state data, whose relevancy was dependent on the ownship state data, would be completely standardized and localized with respect to the ownship.

Individually, each feature underwent an iterative process in order to rank and select best performers. The two features with the highest, individual prediction accuracies were paired together and tested, additional features were added one at a time, to create new feature combinations. The objective was to discover a feature combination that outperformed all the features combined. ML models: Naïve Bayes, Classification Tree, and Discriminant Analysis were used to test the different combinations of features. Results from the testing showed that relative altitude was by far the superior performer, followed by relative altitude rate. This feature pair (relative altitude and relative altitude rate) was designated the *base-pair* from which all other combinations were formulated. The combination of features that performed the best overall are as follows (r*elative altitude, relative altitude rate, angle offset, and aspect angle)* and are ranked according to the order in which they were added to the testing combination. It should be noted that, the relative altitude was squared to eliminate negative data, which rendered a more distinguishable data set by amplifying the difference between aircraft and ownship altitudes. Also, relative altitude-squared (RAS) did not increase overall model accuracy, however, it did increase relevant traffic prediction accuracy to 90% or greater consistently as a single feature input.

As a result four newly derived features were incorporated into the training of TDM-ML: aspect angle, offset angle, relative rate, and relative altitude squared. The aspect and offset angles, were both derived to centralize the course of each aircraft in a traffic scenario with respect to the ownship. The aspect angle provided an accurate depiction of how an aircraft is viewed on the ND. In turn, the offset angle modified the aspect angle in order to directly show how a selected aircraft's course relates to that of the ownship. For example, an offset angle ranged between 0° to 180° indicates that a selected aircraft is on a course which intersects or is parallel to that of the ownship. Incorporating the offset angle provided an instant determination for the vast majority of the time as to convergence of the selected aircraft and the ownship's paths in the horizontal plane. In addition, relative rate offered standardization to the results, providing correlations between surrounding air traffic and the respective ownship.

D. MATIMAL Phase II (MVEL Data Collection Design)

Essential to the application of the ML algorithm was the collection of data that would be used to train the algorithm. For each aircraft, state data was collected via FLTAWR-DG from *Flightaware Live* and segmented into the following predictors: course, airspeed, altitude, vertical trend, range, and bearing, In addition, the following enhanced features: aspect angle, offset angle, relative rate, and relative altitude squared where calculated and made part of the overall list of features used.[2]

Each training scenario was executed with ownship being located differently in the airspace, with the objective of trying to cover as much of a 360° circumference to train the algorithm. The aircraft state data became the core data labels, and subsequently, the enhanced features used to train the TDM-ML algorithm, along with a relevancy marker input by the SMEs specifying each aircraft's importance with respect to ownship. From these features, traffic scenarios, each consisting of 20 aircraft, were created and displayed onto the D.A.T.A ND. Figure 1 depicts the TDM-ML training process and TDM-PM formulation and Figure 2 shows the data collection platform, D.A.T.A.

E. MATIMAL Phase III (Human-Agent V&V Data Collection Design)

The proper training of an agent involves continued V&V of the skills learned from expert instructors. In the case of TDM, its prediction accuracy and how well it performs at determining air traffic relevance could not assessed faithfully by simply applying statistical cross-validation methods. In our case, corroboration from the human-agents it would be teamed with was needed in order to build trust between the agents. Therefore, 28 pilots with expertise in commercial or military aviation were involved in the evaluation of TDM's prediction model. Human-agents validated the accuracy of TDM-PM's relevancy predictions concerning surrounding air traffic with respect to an ownship by affirming or dismissing the relevance of an aircraft's positioning predicted by TDM-PM and then rendered on D.A.T.A.

From the "Pilot Knowledge Data Lake", 0.005% of its remaining data points were used as new input to TDM-PM for the purpose of model testing and pilot/SME V&V. The same 10 scenario screens depicting TDM-PM's prediction of relevant, maybe-relevant, and not-relevant aircraft were presented to each human-agent. The scenarios were not of mixed predictions, but of the same type of TDM-PM predictions, i.e., relevant, maybe-relevant, or not-relevant. Therefore, a scenario screen marked "Relevant Verification" would contain only the relevant predictions of TDM-PM, similarly for the "Maybe-Relevant Verification", and "Not-Relevant Verification" screens. Each air traffic scenario depicted 11 aircraft whose state data elements (altitude, groundspeed, intent, range, and course) were to be compared, by each of the 28 human-agents, to the state data of the ownship. Data analysis of the human-agents V&V of MATIMAL as well as their answers to an end of survey questionnaire are presented in the following results and discussions subsection Phase III.

III. Results and Discussion

In the first generation TDM effort, the TDM-Prediction model was implemented using each of the four supervised learning algorithms individually and compared to each another on performance. MATLAB's TreeBagger algorithm, an implementation of Bootstrap-Aggregated Decision Trees, performed the best at predicting non-relevant data with an accuracy of 89%. However, the other algorithms also showed adequate performances when predicting other relevancies. From this analysis, it was concluded that the second generation TDM-PM was to consider a collection of models capable of capturing separable relationships and phenomena amongst the aircraft and ownship state data.

One of the benefits of utilizing MVEL as the training algorithm was the reduction, and possible elimination, of Type II Error (false-negatives). Due to the safety-critical nature of the research, it is crucial for TDM-PM to have an extremely low rate of incorrectly classifying an aircraft as *"not-relevant"*. Conversely, an incorrect determination of an aircraft's classification as *relevant* does not incur as significant a safety risk as the other classifications. A *not-relevant* or *maybe-relevant* prediction by TDM-PM when an air traffic element was actually relevant was not acceptable, as this posed an unacceptable safety situation.

A. MATIMAL Phase II

Table 1 shows the summary performance assessment of the MVEL models of MVEL. Table 1 describes the split feature set used for each view, before combining them into an ensemble prediction vote. The *"Best Class Predictor"* column describes the *"focus"* of the algorithm in terms of what classification of prediction it performed best at concerning traffic aircraft relevance. The focus was identified when testing numerous feature subset combinations for each algorithm. Relevant and Not Relevant class predictions were the most critical to TDM-PM's prediction accuracy. Therefore, it was necessary to dedicate one view (algorithm) that would best predict a particular classification for each classification focus, hence K-NN for Not Relevant and Naïve Bayes for Relevant. For the remaining two ML algorithms in the MVEL (TreeBagger and SVM), their functionality was not specifically focused to a particular classification or influence, therefore, their TP accuracy and overall accuracy are the same. The *"Best Type True Positive (TP) Accuracy"* column describes the performance of the focus class prediction accuracy, which an individual algorithm was suited best for a certain relevance classification. The "Overall Accuracy" column of the table is a description of each algorithm's prediction accuracy for all relevancy classifications before instituting plurality voting on all of the view's predictions.

Table 1. MVEL Ensemble Model Summary Performances

	Best Class Predictor	Best Type TP Accuracy	Overall Accuracy
K-Nearest Neighbors	**Not Relevant** True-Positive (TP)	**68.8%**	**61.6%**
Naïve Bayes	**Relevant** True-Positive (TP)	**91.7%**	**54.6%**
TreeBagger	All True-Positive (TP)	**62.2%**	**62.2%**
Support Vector Machine	All True-Positive (TP)	**56.1%**	**56.1%**

Table 2, below, contains the analytical results concerning the False-Negative Rate of air traffic relevancies *not-relevant* and *maybe-relevant*. There is no false-negative rate for the *"relevant"* air traffic classification. This is considered a *false-positive*, which is outside the scope of TDM research at this point. The *false-negative* rate for not-relevant air traffic is the number of times that an SME has determined air traffic to be relevant and the TDM-PM has predicted it to be *not-relevant*. Concerning the false-negative rate with respect to maybe-relevant air traffic classification, this represents the number of times air traffic determined *maybe-relevant*, by SMEs, is predicted to be *not-relevant* by TDM-PM. Each row in Table 2 represents the false-negative result of each classification (*maybe-relevant* and *not-relevant*) for each individual MVEL algorithm.

The confusion matrix, Table 3, details the results of the TDM-PM. The feature subset selections for each view in the ensemble was largely influenced by the explored models that had a lower rate of false-negatives and a higher rate of true positives for *not-relevant* and *relevant* aircraft prediction. Within the matrix TDM-PM displays lower false-negatives, which was desirable. The precision column, to the right of the matrix, is the True Positive amount divided by the Sum of the Prediction Conditions. For example, relevant 92.4% is calculated by the relevant true positive divided by the sum of the row: 302 / (302 + 20 + 5). Similarly, for the bottom row of the confusion matrix, often referred to as Recall or Sensitivity, the True Positive amount is divided by the column which is the Sum of the Conditions (actual). For example, *relevant* 52.4% would be: 302 / (302 + 196 + 78). This illustrates that the MVEL had an overall relevant true-positive prediction accuracy of 92.4%, which was much improved over the prediction accuracy of TreeBagger during the development phase of TDM.

Table 2 False-Negative Rate for Not-Relevant and Maybe-Relevant Air Traffic

	Not-Relevant False-Negative Rate	Maybe-Relevant False-Negative Rate
K Nearest-Neighbors	**32** 3.29%	**79** 8.09%
Naïve Bayes	**15** 1.54%	**12** 1.23%
TreeBagger	**18** 1.84%	**92** 9.43%
Support Vector Machine	**11** 1.13%	**43** 4.41%

Table 3 MVEL Prediction Model Error Performance Matrix

	Pilot Relevancy Determinations			
Model Predictions	**Relevant**	**Maybe**	**Not Relevant**	
Relevant	302 30.94%	20 2.05%	5 0.51%	92.4%
Maybe	196 20.08%	98 10.04%	35 3.59%	29.8%
Not Relevant	78 7.99%	61 6.25%	181 18.55%	56.6%
	52.4%	54.6%	81.9%	62.9%

B. MATIMAL Phase III

Verification and validation of the MVEL algorithm was conducted under MATIMAL Phase III, serving three purposes. First, we asked the human-agents to evaluate the performance of the machine-agent (which they actually trained) in determining air traffic relevance. Second, we compared the air traffic relevancy determinations offered by the 28 human-agents between each other

With respect to TDM-PM's performance evaluation by the 28 human-agents, each human-agent's agreement or disagreement with the algorithm's prediction was accessed. Figure 4 indicates that on average the human-agent agreed with TDM-PM's prediction of *relevant* aircraft 41.7% of the time with an average deviation of 12.4%. This was quite compelling, because the human-agents agreed with each other (see Figure 5) 48.7% of the time with a 12.4% average deviation when considering the same *relevant* traffic scenarios. In addition, very similar results were found when considering TDM-PM predictions of *not-relevant* aircraft and human-to-human comparison of the same *not-relevant* aircraft. In those cases the human-agent agreed with the prediction of TDM-PM 58.2% and with each other 65% with average deviations of 25.3% and 24.3% respectively. The greatest disparity between human-to-machine and human-to-human relevancy agreement took place with *maybe-relevant* scenarios. The agreement between human-agents was 24.2% greater than their agreement with TDM-PM. This can be attributed to the fact that 61% of the human-agents' V&V of TDM-PM's maybe-relevant predictions were consider not-relevant. Overall, the assessment of TDM-PM's ability to predict air traffic relevancy effectively is in need of improvement, which can only be realized through continued training with relevant and maybe-relevant scenarios. However, at TDM's current stage of development, its ability to predict relevant air traffic is comparable to that of human-agents. This is significant when addressing the increase of autonomous systems in the flight deck of aircraft and when considering how to develop a synergy between IAS and the human-agents that will be training it, because it identifies a training procedure that would lend itself to both improving the prediction accuracy of the machine-agent an provide a relationship of trust between it and the human-agent.

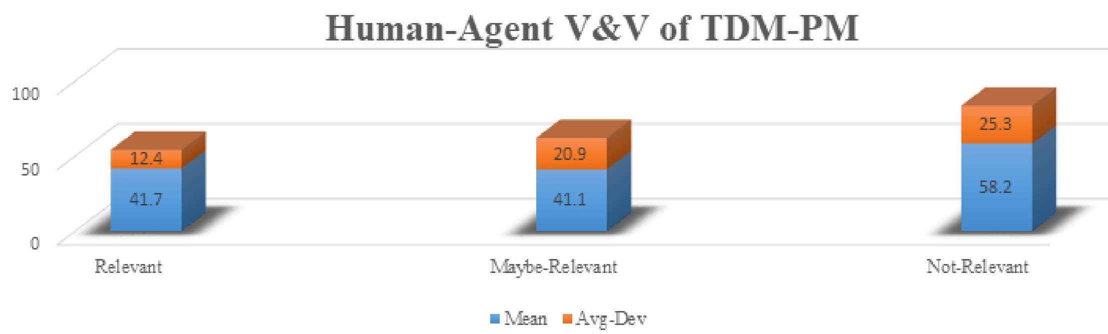

Figure 4. Pilot Agreement of TDM-PM Air Traffic Relevancy Predictions

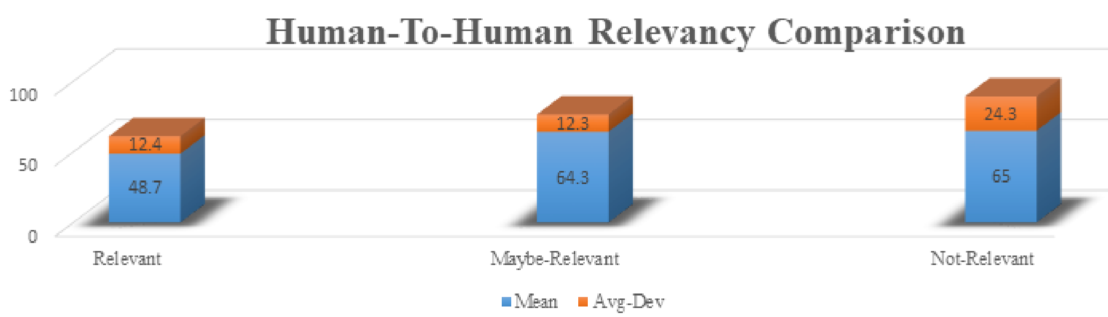

Figure 5. Human-Agents of Aircraft Relevancy Determinations

IV. Conclusion

During MATIMAL Phase II, the overall objective for the TDM-PM, machine-agent, was to improve its prediction accuracy for classifying relevant data. Additionally, it was a high priority to better define and understand the data prediction boundaries. Unlike the first generation TreeBagger TDM-ML model, the second generation data was subsampled to have an equal size dataset for each of the relevancy classes. The same supervised learning algorithms (K-NN, Naïve Bayes, Support Vector Machines, and TreeBagger) were explored using the equal-size data bins for each class. Although the models resulted in having more balanced prediction accuracies than the first generation models, the prediction accuracies were slightly lower. After further exploration of the data through feature selection and feature analysis, it was found that model combinations of the classifiers tended to perform better than all of the standalone models.

Having established a MVEL that infers, with very good accuracy, relevant classification of air traffic with respect to an ownship, the next pursuit was finding the same type of algorithm to improve maybe-relevant and not-relevant classifications. Ensembling the three MVELs together formed a ML and prediction model that covers, with high accuracy, all three categories of air traffic relevance. Our next step was a final verification and validation of the classifications using focused MVEL ensemble, by applying a conformational approach, having SMEs confirm the model's aircraft relevancy prediction accuracy.

In MATIMAL Phase III, human-agent verification and validation of TDM-PM focused on the HAT that needs to be developed in order for the integration of IAS in the cockpit to become more than a theoretical prospect. This was done by allowing the human-agent that trained the machine-agent to assess its accuracy level and offer suggestions at improving the algorithms capabilities.

Because of the noticeable parity in human-agent determination of and machine-agent prediction of air traffic relevance, a training scheme was offered as a way to build proper HAT and improved training for the IAS. Once the machine-agent has initially gone through the Supervised Machine Learning Workflow process, see Figure 1, then the same flight crew for training is paired to the IAS and remains through the evaluation process. This will allow the machine-agent the opportunity to learn the nuances and idiosyncrasies that influenced the IAS accuracy in determining relevance. It will also build the symbiotic relationship, and trust through verification, between human and machine that is necessary in order for IAS to become a reality in the cockpit. Post research survey questions suggest positive feedback toward that proposal. There was an 87.0% approval rating from human-agents when asked of the usefulness of an algorithm with air traffic partitioning. When asked which agent (human or machine) would be more accurate at predicting air traffic relevance 78% of the subject matter experts suggested that the machine-agent would be better suited for that if properly trained. With a 90% consensus, the human-agents felt that the machine-agent's true positive prediction rate should be 90% and there was also the belief that there human counterparts performing the task of determining air traffic relevance were 89% accurate on average. However, their agreement with fellow pilots was comparably the same as it was for machine-agents, which is represented in Figures 5 and 6, where there is only a 7% difference in relevant and not-relevant machine-agent predictions and human-agent determinations and 23% difference in maybe-relevant machine-agent predictions and human-agent determinations.

Acknowledgements

Authors' thanks go to Randy Bailey (Flight Deck Interface Technology Team Lead), Jason R. Neuhaus (Senior Aerospace Engineer at NASA Langley Research Center), Trey Arthur (Senior Research Engineer at NASA Langley Research Center), Steve Williams (Senior Research Engineer at NASA Langley Research Center), and the entire Flight Deck Interface Team.

References

[1] FAA Aerospace Forecast Fiscal Years 2009 – 2025. Retrieved at: https://www.faa.gov/data_research/aviation/aerospace_forecasts/2009- 2025/media/2009%20Forecast%20Doc.pdf

[2] Houston, V, E., Le Vie, L, R., "Autonomous System Technologies for Resilient Airspace Operations", *AIAA Aviation Conference*, Denver, CO, 2017

[3] Kumar, V., Minz, S., "Multi-view ensemble learning: an optimal feature set partitioning for high-dimensional data classification", Springer-Verlag London , [published online] 21September 2015, URL: https://link.springer.com/content/pdf/10.1007%2Fs10115-015-0875-y.pdf

[4] NASA Aeronautics Airspace Operations and Safety Program Discussion with the ARTR, see: http://sites.nationalacademies.org/cs/groups/depssite/documents/webpage/deps_152342.pdf

[5] Bishop, C, M., "Pattern Recognition and Machine Learning", University Course Book February 2006

[6] Mathworks MATLAB TreeBagger Class, https://www.mathworks.com/help/stats/treebagger.html

[7] Bishop, C, M., "Neural Networks for Pattern Recognition", University Course Book 1995

[8] Mathworks MATLAB SVM Class, see: https://www.mathworks.com/help/stats/compactclassificationsvm.predict.html?searchHighlight=SVM&s_tid=doc_srchtitle#bt78_e8