

Article

Robust and Explainable Semi-Supervised Deep Learning Model for Anomaly Detection in Aviation

Milad Memarzadeh ^{1,*} , Ata Akbari Asanjan ¹ and Bryan Matthews ²

¹ Universities Space Research Association (USRA), Data Sciences Group, NASA Ames Research Center, Moffett Field, CA 94035, USA

² KBR Inc., Data Sciences Group, NASA Ames Research Center, Moffett Field, CA 94035, USA

* Correspondence: milad.memarzadeh@nasa.gov

Abstract: Identifying safety anomalies and vulnerabilities in the aviation domain is a very expensive and time-consuming task. Currently, it is accomplished via manual forensic reviews by subject matter experts (SMEs). However, with the increase in the amount of data produced in airspace operations, relying on such manual reviews is impractical. Automated approaches, such as exceedance detection, have been deployed to flag safety events which surpass a pre-defined safety threshold. These approaches, however, completely rely on domain knowledge and outcome of the SMEs' reviews and can only identify purely threshold crossings safety vulnerabilities. Unsupervised and supervised machine learning approaches have been developed in the past to automate the process of anomaly detection and vulnerability discovery in the aviation data, with availability of the labeled data being their differentiator. Purely unsupervised approaches can be prone to high false alarm rates, while a completely supervised approach might not reach optimal performance and generalize well when the size of labeled data is small. This is one of the fundamental challenges in the aviation domain, where the process of obtaining safety labels for the data requires significant time and effort from SMEs and cannot be crowd-sourced to citizen scientists. As a result, the size of properly labeled and reviewed data is often very small in aviation safety and supervised approaches fall short of the optimum performance with such data. In this paper, we develop a Robust and Explainable Semi-supervised deep learning model for Anomaly Detection (RESAD) in aviation data. This approach takes advantage of both majority unlabeled and minority labeled data sets. We develop a case study of multi-class anomaly detection in the approach to landing of commercial aircraft in order to benchmark RESAD's performance to baseline methods. Furthermore, we develop an optimization scheme where the model is optimized to not only reach maximum accuracy, but also a desired interpretability and robustness to adversarial perturbations.



Citation: Memarzadeh, M.; Akbari Asanjan, A.; Matthews, B. Novel Methods for the Global Synchronization of The Complex Dynamical Networks with Fractional-Order Chaotic Nodes. *Aerospace* **2022**, *9*, 437. <https://doi.org/10.3390/aerospace9080437>

Academic Editor: Gokhan Inalhan

Received: 23 June 2022

Accepted: 8 August 2022

Published: 10 August 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: anomaly detection; semi-supervised learning; explainable AI; aviation safety

1. Introduction

In real world applications, obtaining high quality labels for a dataset is a challenge that requires significant effort from subject matter experts (SMEs). On the other hand, supervised learning methods (the most commonly used method in the data science and machine learning domains) require fully labeled datasets to train a model that generalizes well on unseen data. Depending on the application and the complex nature of the data, a subset of data can be reviewed and labeled by the SMEs. However, if the data is exceedingly complex, it takes significant amount of time to label them, and as a result, a high number of labeled examples can be unrealistic to obtain. Furthermore, advanced machine learning models (e.g., deep learning) can contain an extremely large number of trainable parameters. This requires a significant amount of labeled data to reach optimum performance and prevent the model from overfitting. Techniques such as crowd sourcing, a term coined in 2005 [1], can help alleviate this problem, constrained by the "crowd's" skill level. If the objective is to label images of dogs and cats, the available pool of labeling workforce can be

broad since the majority of the population can easily identify these animals in a variety of photographic conditions. Other applications may require some degree of training, and this can be incorporated into the labeling process. In such setting, a level of trust or confidence based on monitoring a user's known experience can be assigned for each label. This helps track the quality of the labels such as in the Nemo-Net [2] and Galaxy Zoo [3] citizen science projects. While this works well for intermediate levels of expertise, in domains requiring more specific and technical knowledge, the pool of SMEs with relevant background and experience to validate each data instance may be drastically decreased. Furthermore, the data might be considered proprietary and sensitive (as is the case in the aviation safety domain) and hence cannot be shared with citizen scientists.

The lack of available SMEs is particularly compounded in aviation safety by the fact that not all events that occur are purely threshold crossings. They may involve reviewing a variety of actions, conditions, and sequences of events corroborated across various data sources before it can qualify as an operationally significant event. As a result, to properly assign the data labels, a forensic analysis for an hour of review may only yield a dozen or so labels [4]. Moreover, an increasingly important area of focus in this domain is vulnerability discovery or anomaly detection—especially detection of unknown anomalies. In this case, there is no pre-defined notion of the anomalies that the algorithm has learned to detect. These types of anomalies also add time to the review process because there may not be precedent to categorize these anomalies. In addition to this, it may take more time to fully consider the scenario's significance, its proximity to the safety margins, and the impact it may have had on the operations.

In the realm of machine learning algorithms that can be used to address this problem, there exist three main approaches: unsupervised, supervised, and semi-supervised learning. Purely unsupervised approaches can be prone to high false positive rates and do not perform as well as supervised approaches when labels are present. On the other hand, as discussed above, a completely supervised approach might not reach optimal performance when the size of labeled data is small. Furthermore, any supervised model can only detect known (labeled) anomalies and hence suffer from the inability to discover unknown vulnerabilities. Semi-supervised learning combines the two approaches and has the ability to leverage the known labeled data as well as the patterns found in the vast pool of available unlabeled data [5]. This gives the semi-supervised approaches an advantage that can potentially address drawbacks of both unsupervised and supervised models, especially those associated with the lack of labeled data (when only a small set of labels can be acquired).

In addition to the performance improvements from semi-supervised learning where both the labeled and unlabeled data are leveraged, the algorithm yields useful structuring of the data in the learned feature space (i.e., latent space). By examining labeled examples that are collocated within this arrangement of data, it can be inferred that the unlabeled data instances have similar characteristics. This provides some level of model explainability in how the data is organizing in this space and can be used to find other similar events within the unlabeled data.

In the aviation safety domain, cases such as unstable approaches [6] are well defined and use threshold crossings to detect the occurrence and level of severity for known events. In this paper, we leverage these known anomalies from commercial aircrafts' approach to landing data to help bootstrap and guide a novel semi-supervised learning approach. We benchmark performance of the model based on three synergistic criteria: (1) classification accuracy, (2) interpretability of the extracted/learned features, and (3) robustness to adversarial perturbations. These criteria are compared against several baseline methods in aviation and machine learning literature. Furthermore, as we will discuss later in the results section, the similarities among the extracted features from the data can be exploited to help identify samples that may contain previously unknown anomalies by having the SMEs in the loop. This sets out the next step to design an active learning mechanism, where these newly discovered anomaly categories can be refactored into the training set. This allows us

to begin to build previously unknown classes in the data, and further guide the algorithm's ability to classify the categories and discover new anomalies over subsequent iterations.

In a deployed operational environment, imagine a situation where borderline events that may not meet the strict criteria currently being used to define an event, are labeled nominal and are therefore unmonitored. This approach can assist in identifying these borderline events since they share similar patterns with the labeled events. Uncovering and understanding an expanded events category can provide operators a clearer assessment to their risk exposure. This can also offer a means for crafting more accurate event definition logic that encompasses the more comprehensive event category. In other words, this technique can add new insights into the modes of operation that previously had been limited in scope and help improve the overall safety of the operations.

2. Related Work

Machine learning has been widely applied in the aviation domain for anomaly detection and safety improvement applications [7–10]. Due to the lack of properly labeled aviation data, the majority of the literature on aviation safety and anomaly detection has focused on unsupervised learning [11]. Unsupervised approaches can roughly be categorized into: (1) distance-based [12–15], (2) kernel-based [9], and (3) deep learning-based methods [11,16]. Distance-based models (nearest neighborhood and clustering approaches) are proven effective by using a distance metric to identify anomalous events. This category of models, however, is less popular due to their quadratic computation complexity [11]. Bay and Schwabacher [12] made one of the early attempts to reduce the computational complexity of distance-based anomaly detection, defining anomalies as points with far-off nearest neighbors. Kernel-based approaches such as One-Class Support Vector Machine (OC-SVM) are frequently used for unsupervised anomaly detection applied to the aviation domain. NASA's Multiple Kernel Anomaly Detection (MKAD) model [9] uses OC-SVM as a part of the overall framework and demonstrates significant proficiency in finding operationally significant anomalies in heterogeneous time-series of commercial flights.

More recently, deep learning has become popular in anomaly detection literature. Specifically, methods based on deep generative models such as Auto-Encoders (AEs) [17], Variational Auto-Encoders (VAEs) [18], and Generative Adversarial Networks (GANs) [19] have been widely adopted for anomaly detection purposes in many science and engineering domains. A popular subset of these approaches is reconstruction-based anomaly detection. In this approach, a deep generative model is used to reconstruct/generate the input data by sampling from a lower-dimensional latent feature space. The main intuition is that since the majority of the training data are nominal, the reconstruction error for those data would be lower compared to the minority anomalous data present in the training. In other words, reconstruction-based models bet on the noticeable inconsistency of anomalies in subspace representation resulting in high reconstruction errors. This approach has been widely used for identifying anomalies in time-series data [20–25] as well as aviation data [7,16,26,27]. Janakiraman and Nielsen [7] have implemented an extreme learning models AE to learn the nominal distribution. The anomalies are predicted based on surpassing the reconstruction error for a nominal boundary. Wang et al. [26] developed a transfer learning-based AE that forces the latent space to learn useful data aspects. The authors applied this model to flight track anomaly detection problems on data from multiple airports and reported high performance and high capability for the model to reduce data processing requirements. Memarzadeh et al. [16] have developed a Convolutional Variational Auto-Encoder (CVAE) to detect flight track anomalies. The model used an ℓ_2 distance reconstruction error as a metric to identify anomalies.

Despite the compelling case of no labeling requirement for unsupervised approaches, their performance is not competitive compared to supervised models. Lee et al. [28] introduced a framework called Safety Analysis of Flight Events using classic supervised machine learning models. They showcased the versatility of the framework using Flight Operational Quality Assurance (FOQA) data in identifying multiple anomalies in the approach to landing of commercial aircraft. In another study, Janakiraman [29] developed a supervised

precursor mining algorithm: Deep Temporal Multiple Instance Learning (DT-MIL). This algorithm finds anomalies by correlating incoming events to anomalous multi-dimensional time-series. The author used deep recurrent neural networks in a multi-instance learning structure to efficiently track temporal behavior. Despite the significant predictive capabilities of supervised methods, developing such an approach can be expensive and infeasible at times. This is especially true in aviation safety datasets, since acquiring reliable and accurate labels for data requires significant time and effort from SMEs and is largely impractical. On the other hand, unsupervised methods are cheaply available, so long as you assume an operationally significant aviation anomaly is equivalent to a statistical one. This assumption is not consistently correct and results in poor performance of unsupervised approaches compared to supervised methods in application to the aviation domain. This leads to a significant number of false positives (false alarms) that bring into question the reliability and applicability of unsupervised methods.

There are limited studies in aviation safety literature to fill the gap between unsupervised and supervised methods. Active learning [30–33] has been developed to tackle this problem, where different information-theoretic or uncertainty-based methods are used to identify the most informative data (among the vast pool of unlabeled data) to be reviewed and labeled by SMEs. Although this approach improves the performance and efficiency of the supervised methods (by incorporating the smart labeling strategies into account), it does not tackle the shortcomings of the supervised learning approaches completely and still requires SMEs in the loop.

Semi-supervised methods are potential approaches to fill the gap where the labeled data exist but are not sufficient for fully supervised modeling. These approaches have been applied in the anomaly detection of time-series [34,35]; however, they have not yet been truly explored in the aviation safety domain. To our knowledge, the only existing semi-supervised aviation anomaly detection is a study done by the authors of this paper [36] where two recent semi-supervised approaches have been used for detection of aviation anomalies.

3. Method

In this paper, we develop RESAD, a Robust and Explainable Semi-supervised deep learning model for Anomaly Detection in aviation data that addresses the shortcomings of both supervised and unsupervised learning. The semi-supervised mechanism allows the decision makers to make inference based on minimally available (but extremely valuable) labeled data as well as the vast amount of unlabeled data. As a result, it overcomes the main disadvantages of these two families of methods: (1) supervised learning not performing optimal due to scarcity of labeled data, and (2) unsupervised learning not showcasing great accuracy and reliability by not leveraging operational domain knowledge from SMEs. The proposed semi-supervised model is also superior to active learning as it does not rely on the availability of SMEs for data labeling; however, it can be easily fit within an active learning framework.

We build the model upon two existing methods in machine learning literature [37,38], and show that it is superior to multiple baseline methods from literature in flight multivariate time-series anomaly detection. Specifically, we train RESAD based on a loss function that: (1) takes advantage of both labeled and unlabeled sets of data to extract informative features for accurate classification of multi-classes of anomalies; (2) uses graph theory-based label propagation and enforces a compact clustering of data belonging to each class in the latent feature space, which improves the interpretability of extracted features and its application for down-stream tasks; and (3) uses the reconstruction fidelity of the input data based on its generative capability to improve the robustness of the learned latent features to adversarial perturbations.

Let us imagine that the available data is grouped into two sets: the minority labeled set, (X_L, y_L) , and the majority unlabeled set, X_U , where the size of the unlabeled set is significantly larger, i.e., $|X_U| \gg |X_L|$. It should be noted that any supervised learning technique would ignore X_U , while any unsupervised learning method would ignore y_L .

As depicted in Figure 1, RESAD consists of three components: (1) an encoder, (2) a decoder, and (3) a classifier. The encoder, $q_\phi(z | x)$, is a deep convolutional neural network (exact architectures are reported in Appendix A) that maps the input data X to a latent feature space Z . The decoder, $p_\theta(x | z)$, is also a deep convolutional neural network that reconstructs the data \hat{X} from the latent features Z . The classifier, $c_\psi(y | z)$, is a fully connected neural network with dropout regularization that classifies the data in the latent feature space. Parameters ϕ , θ , and ψ represent the weights of the neural network for the encoder, the decoder, and the classifier, respectively.

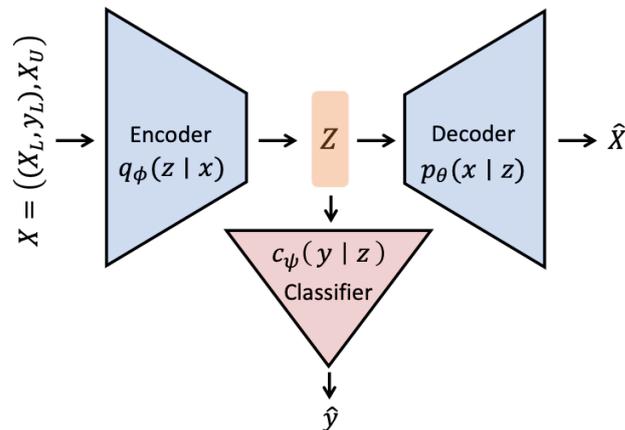


Figure 1. Graphical illustration of RESAD's architecture and its components.

We train the entire network end to end and use all available data (labeled and unlabeled). The overall objective of the optimization is to find a set of weights (i.e., ϕ^* , θ^* , ψ^*) that minimizes the following loss function:

$$\mathcal{L} = w_s \mathcal{L}_{cls} + w_c \mathcal{L}_{cclp} + w_r \mathcal{L}_{rec} \quad (1)$$

The first term is the classification loss and is defined as a cross entropy (\mathcal{H}) between the prediction of classifier and the true labels on the labeled set:

$$\mathcal{L}_{cls} = \mathbb{E}_{(X_L, y_L)} [\mathcal{H}(y_L, c_\psi(y | q_\phi(z | X_L)))] \quad (2)$$

The second term in Equation (1) corresponds to the compact clustering via label propagation (CCLP) loss. We have adopted this loss term from [38] and it is defined as follows:

$$\mathcal{L}_{cclp} = \mathbb{E}_{Z \in \{Z_L \cup Z_U\}} \left[\frac{1}{S} \sum_{s=1}^S \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N -T(z_i, z_j) \log H^{(s)}(z_i, z_j) \right] \quad (3)$$

where $N = |X_L| + |X_U|$ is the total number of training data, T is the optimal transition matrix between data instances in the latent feature space, H is the actual transition matrix estimated via dynamic graph construction and label propagation, and S is the step of the Markov chain on the graph. Equation (3) is the cross-entropy between the desired optimal transition function T and the estimated one H .

To estimate H , we first calculate the adjacency matrix, A , which is estimated based on the similarity of the data instances in the latent space. We define the adjacency matrix as follows using Cosine similarity,

$$A(z_i, z_j) = \exp(z_i \cdot z_j^T) \forall z_i, z_j \in \{Z_L \cup Z_U\} \quad (4)$$

where T is transpose operation. It should be noted that the results are not affected by the choice of similarity measure, and any other metric (such as negative Euclidean distance) can also be used as a similarity metric. The Markovian random walk along the nodes of

this graph is defined by the transition matrix H , which is obtained by row-normalizing the adjacency matrix A ,

$$H(z_i, z_j) = \frac{A(z_i, z_j)}{\sum_k A(z_i, z_k)} \quad (5)$$

Once the graph is constructed according to the transition matrix H , label propagation uses H to propagate the class confidence from the labeled to unlabeled samples and estimate the optimal transition function T . This is an iterative process until the process converges at an equilibrium. The class posteriors for the unlabeled data, Φ_U at this equilibrium can be computed in closed form [38] as follows,

$$\Phi_U = (\mathbf{I} - \mathbf{H}_{UU})^{-1} \mathbf{H}_{UL} \mathbf{Y}_L \quad (6)$$

where H is re-arranged to its labeled and unlabeled elements as follows,

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{LL} & \mathbf{H}_{UL} \\ \mathbf{H}_{LU} & \mathbf{H}_{UU} \end{bmatrix} \quad (7)$$

As a result, $\Phi = \begin{bmatrix} \mathbf{Y}_L \\ \Phi_U \end{bmatrix} \in \mathbb{R}^{N \times n_c}$ is the class posterior estimated by the label propagation at convergence, and n_c is the number of known classes.

Finally, the optimal transition function between data instances, T , is calculated based on the class posterior, i.e., Φ . The equilibrium denotes an optimal state in which transition probability between any two data instances of the same class is the same, and it is zero for inter-class transitions. Kamnits et al. [38] provides the following formula for calculating this optimal transition function,

$$T(z_i, z_j) = \sum_{c=1}^{n_c} \phi(z_i, z_c) \frac{\phi(z_j, z_c)}{m_c}, \quad m_c = \sum_{i=1}^N \phi(z_i, z_c) \quad (8)$$

where $\phi(z_i, z_c)$ is the posterior for node i to belong to class c , and m_c is the expected mass assigned to class c .

Finally, the third term in Equation (1) is the reconstruction loss, which ensures that the latent feature space is informative and robust enough that we can reconstruct the input data accurately from it. We define the reconstruction loss as the binary cross entropy (BCE) between the input data and the reconstruction; however, any other metric such as mean squared error (MSE) can be used as well. Our experiments have shown that when the input data is normalized using MinMax scaling, meaning that all the features take ranges between 0 and 1, the time-step level (or pixel-level in case of imagery data) BCE loss captures the variability in the reconstructed data compared to the input data much better than the MSE loss. However, it should be noted that if the data is scaled using standard scaling, and, as a result, features are not necessarily bounded between 0 and 1, only MSE loss should be used. The reconstruction loss is formalized as follows:

$$\mathcal{L}_{rec} = \mathbb{E}_{X \in \{X_L \cup X_U\}} [\mathcal{H}(X, p_\theta(X | q_\phi(Z | X)))] \quad (9)$$

w_s , w_c , and w_r in Equation (1) are the hyper-parameters that tune the importance of each loss term in the overall loss function.

4. Results and Discussion

We compare the performance of RESAD with two baseline semi-supervised models: (1) Compact Clustering via Label Propagation (CCLP) [38] and (2) Auto-Encoder + Classifier (AE+C) [37]. Kingma et al. [37] proposed a generalization of deep generative models such as VAEs (which is a widely used deep learning method for representation learning, nonlinear dimensionality reduction, and anomaly detection in many domains and in our previous work [16]) to a semi-supervised version by adding a classifier to the VAE structure that is mainly trained based on the minimally available labeled data. They showed that such a

semi-supervised deep generative model is superior to classic semi-supervised methods such as transductive support vector machines [39], especially when the size of unlabeled data is huge. Later, Kamnitsas et al. [38] developed the CCLP formulation, a discriminative model with a novel cost function for semi-supervised learning based on deep learning and graph theory, and showed that it is superior to developed architectures based on deep generative models such as [37]. These two models can be seen as simpler models compared to ours. CCLP only contains an encoder and classifier and is trained only based on the first two terms in Equation (1), while AE+C has an encoder, decoder, and a classifier, but does not enforce compact clustering, and is only trained based on the first and the third terms in Equation (1).

We quantify the comparison according to three metrics: (1) classification performance, (2) latent space configuration, and (3) robustness to the adversarial perturbations. For classification performance, we also include a comparison with DT-MIL [29] to show the superiority of semi-supervised learning over supervised methods when the labeled data is scarce. DT-MIL is chosen as the most recent supervised anomaly detection model based on deep learning architecture that has been validated on the FOQA data (similar data that has been used in this study). For the second metric, we qualitatively and quantitatively show how interpretable and useful the learned features in the latent space are for the downstream tasks (e.g., active learning, clustering). Lastly, the third metric evaluates the robustness of the inference made by the models to noise and adversarial perturbations in the input data.

The next subsection describes a real-world multi-class anomaly detection dataset during approach to landing of commercial aircraft. We have developed this dataset to benchmark performance of our proposed method, i.e., RESAD, against the baseline methods mentioned above.

4.1. Multi-Class Anomaly Detection during Approach to Landing of Commercial Aircraft

In this section, we introduce a multi-class anomaly-detection dataset based on FOQA data from a commercial airline <https://c3.nasa.gov/dashlink/projects/85/> (accessed on 1 March 2021). This data is primarily comprised of 1-Hz recordings for each flight and covers a variety of systems. These include the state and orientation of the aircraft, positions and inputs of the control surfaces, engine parameters, autopilot modes, and corresponding states. The data is acquired in real time on-board the aircraft and downloaded by the airline once the aircraft has reached the destination gate. These time series are analyzed by SMEs to flag known events and create labels. Each data instance is a 160-second-long recording of 20 variables during the approach of the aircraft to landing—from a few seconds before an altitude of 1000 ft, to a few seconds after an altitude of 500 ft. It should be noted that, for many flights, depending on the landing runway and airport geometries, the duration from 1000 to 500 ft altitude is less than 160 s. In this case, we expand the data window to include an additional period directly before reaching 1000 ft altitude.

We processed and labeled 30,522 overall data instances, which is comprised of four classes: (1) nominal, where no anomaly of the other three classes is known to be present (~66.7% of the total data); (2) speed anomaly, where the anomaly is identified based on a deviation from the target landing airspeed during approach (~22.9% of the total data); (3) path anomaly, where the path of descent for landing are flagged as being anomalous and deviated significantly from the glide slope (~7.2% of the total data); and (4) control anomaly, where the flaps (specific control surface on the wings of the aircraft) are flagged anomalous if there is a delay in extension as compared to the expected nominal deployment during approach to landing (~3.2% of the total data). These events were chosen because they are all relevant metrics used to measure unstabilized approaches. Figure A4 in Appendix C visualizes the flight time-series in the training set in 2D using t-Stochastic Neighbor Embedding (t-SNE) [40] color-coded based by their true class. It appears that there are some distinct modes/clusters in the input space, but none are corresponding to the known classes of anomaly present in the data. This makes the task of anomaly detection in the input space difficult, since the data is not easily separable and organized. We aim to utilize our proposed semi-supervised method, RESAD, to efficiently generate latent

representations with easily separable boundaries and with an interpretable configuration that down-stream tasks can leverage.

Each data instance is either nominal or contains only one type of anomaly: a restriction that simplifies the validation process. Testing on data that contains multiple types of anomalies per instance will be part of our future work. Figure 2 shows the distribution of the data based on the landing airport. As it can be seen, majority of the data is for landing in Minneapolis–Saint Paul International Airport, Detroit Metropolitan Wayne County Airport, and Memphis International Airport.

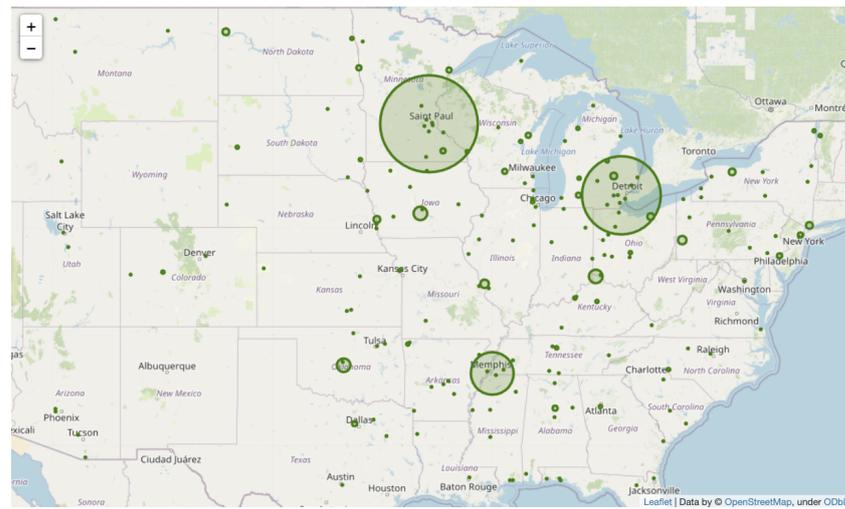


Figure 2. Distribution of the data across landing airports. The sizes of the circles are proportional to the amount of data for each airport.

We divide the data into three sets of training (60%), validation (20%), and testing (20%). The training set is used for training the models. The validation set is used to select an optimal choice of hyper-parameters (discussed in the next section). The testing set is used to report an unbiased estimate of the models' performances. All the figures presented throughout the paper are based on the results obtained by applying the models to the testing set (an unseen set during training, validation, and hyper-parameter tuning).

4.2. Implementation Details

All three semi-supervised models are implemented in Python using the PyTorch library. The architecture of the encoder, the decoder, and the classifier in all of them are identical and are reported in Appendix A. The DT-MIL model was used the way it was originally implemented, using the Keras library (for details refer to [29]). DT-MIL is a binary classification model. Since we intended not to alter the original model, we implement DT-MIL as a one-versus-all scheme for the case of multi-class classification.

All models were trained for 200 epochs; the Adam optimizer [41] was used for all models with a learning rate equal to 3×10^{-4} and default momentum parameters. We performed hyper-parameter tuning based on the validation set to identify reasonable choices for the hyper-parameters. Based on our comprehensive experimentation, the latent space dimension of all three semi-supervised models is fixed to 256 dimensions ($Z \in \mathbb{R}^{256}$), the number of steps in the Markov chain in Equation (3), S is fixed to 3, and the weights of different loss terms in Equation (1), i.e., $\{w_s, w_c, w_r\}$ are fixed based on the size of the labeled set, i.e., $|X_L|$. We report these values in the next section, where we discuss the findings.

4.3. Classification Performance

Figure 3 compares performance of the classification in terms of average accuracy (mean \pm standard deviation) of classification among our proposed model, RESAD, (green) with the baseline models. This is based on 20 independent trials of training, where

the labeled set is sampled uniformly across classes and randomly within each class. For example, in the case of a 100-sample labeled set, 25 samples are randomly selected from each of the four classes. The x -axis shows the number (and percentage) of the labeled set in the training. It should be noted that the results presented in the figure are based on the performance on the testing set that was not seen by the algorithms in either training or validation phases (hyper-parameter tuning). As mentioned before, we set aside a validation set to perform hyper-parameter tuning and find the right combination of weights in the loss function in Equation (1) for our approach. Based on our comprehensive experimentation, the following general rules emerged: for the classification loss (the first term), we found that having a higher weight for the case of small labeled set (e.g., 100 and 200 samples) improves the performance, while a small weight might be sufficient when the size of labeled set grows (e.g., 500 and 1000 samples). A large weight for CCLP loss (the second term) was found to improve the performance. The weight of the reconstruction loss (the third term) did not play a major role in the classification task, but was very crucial in the robustness to the adversarial perturbation (we will discuss this later in this section). Based on the experimentation, we fixed the values of the weights to $w_s = 100$ for $N_L \in \{100, 200\}$ and $w_s = 1$ for $N_L \in \{500, 1000\}$, $w_c = 100$, and $w_r = 10$. It should be noted that we performed hyper-parameter tuning only for our proposed model. For the baseline methods, we kept the loss function of training and corresponding weights of the terms identical to the ones obtained by the original authors.

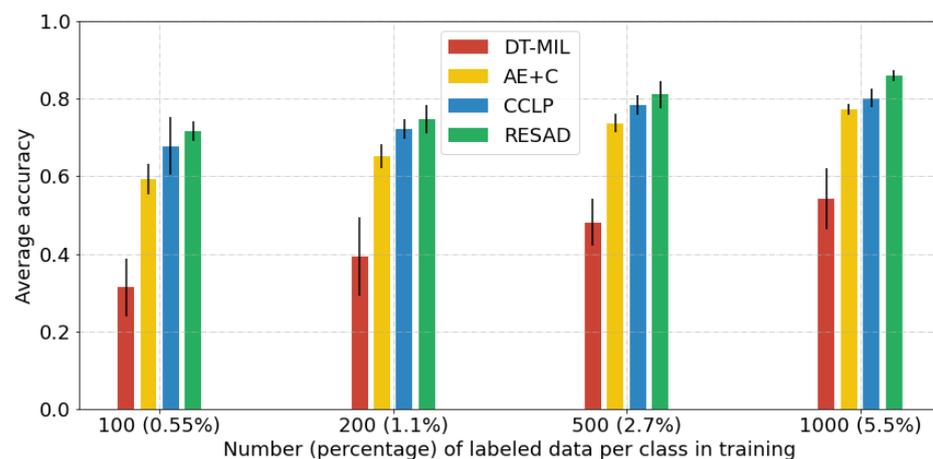


Figure 3. This figure compares performance (average accuracy of classification) of our proposed model (colored green) with multiple baseline methods, such as DT-MIL (red), AE+C (yellow), and CCLP (blue).

A major finding in Figure 3 is that all the semi-supervised models significantly outperform the supervised DT-MIL method. This emphasizes the superiority of the semi-supervised learning when the size of the labeled set is small. Among the semi-supervised models, RESAD performs slightly better than the baseline models. Figures A5 and A6 in Appendix C show the precision and recall values per class for each method. Semi-supervised methods perform significantly close in recall of identifying anomalous classes. RESAD has a higher precision in the minority anomaly classes (path and control in Figure A5) and a higher recall for the majority nominal class (Figure A6).

Furthermore, we calculate precision, recall, F1-score, and AUROC (Area Under ROC curve) for the binary anomaly detection problem, where we evaluate how accurately the model can distinguish between nominal versus anomalous classes. These performance metrics are reported in Table A1 in Appendix C. Although the difference between the semi-supervised models might not seem significant in the classification performance, we shall see later that the differences are significant with respect to other metrics.

4.4. Latent Space Configuration

Figure 4 visually compares the configuration of the latent feature space for the three semi-supervised models, i.e., AE+C, CCLP, and RESAD. All of the figures are showing the best example out of the 20 independent trials of training for each model, based on a 1000-sample labeled set. All figures are visualizing the 256-dimensional latent feature space of each method in 2D using t-SNE, initializing it with Principal Component Analysis (PCA), and setting the perplexity parameter to 50. The left column color-codes the data instances based on the true class that they belong to (blue: nominal, orange: speed anomaly, green: path anomaly, and red: control anomaly). As it can be seen, RESAD and CCLP show significant improvement over the AE+C in compactly clustering the data of each class together and far away from other classes. This distinction is less realized in the AE+C approach. This is an intuitively justified result, since both CCLP and RESAD use graph theory to enforce such compact clustering in the latent feature space, while AE+C does not enforce that.

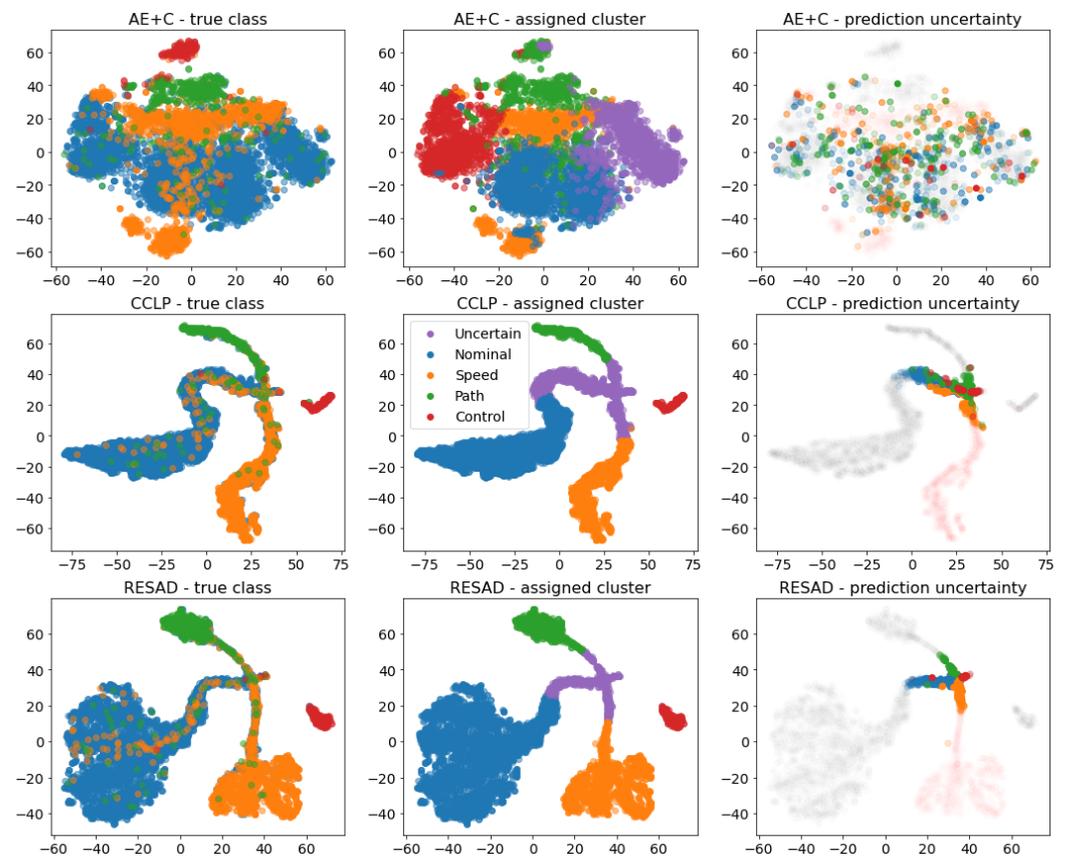


Figure 4. This figure visualizes an example of the learned latent feature space for AE+C, CCLP, and our approach. The left column is color-coded based on the actual class that each data instance belongs to, the middle column is color-coded based on the clusters found in the latent feature space, and the right column illustrates data instances that the classifier has the highest amount of uncertainty in classifying them. All plots are illustrating the 256-dimensional latent space in 2D using t-SNE.

To both make sure that this compact clustering is not an artifact of t-SNE's nonlinear embedding from 256D to 2D, as well as to understand the structure and configuration of the latent feature space better, we perform unsupervised clustering in the 256D latent space of these models. The middle column in Figure 4 shows the result of applying KMeans clustering with $K = n_c + 1$ (n_c being the number of classes in the training data) and Euclidean distance as a distance metric in the 256D latent space, visualized in 2D using t-SNE. In order to associate each cluster with true classes, we use the classifier's prediction for the data in the cluster. For example, if the majority of the data in the cluster are classified

as nominal, we associate that cluster with the nominal class. We use the color purple to show the $(n_c + 1)$ -th cluster (fifth cluster in here), which we call the uncertain cluster. As illustrated, both CCLP and RESAD confirm that data of each class are compactly clustered together, while they are far away from data of other classes, with a central cluster (the purple cluster) merging them together. However, we can see that in the case of AE+C approach, the clusters that are found using KMeans do not necessarily correspond to the actual classes of the data. For example, we can see that the data of both path and control anomalies are clustered together (green cluster in the middle column, top panel), which is not a desired structure of the latent feature space. Moreover, the data of the nominal class is clustered into multiple smaller ones, and the purple cluster does not play a role of central merging cluster between different classes of the data.

Right column of Figure 4 visualizes data instances, where the classifier has the highest amount of uncertainty in classifying them. In order to quantify the uncertainty of the classifier's prediction, we use the entropy of the output of the Softmax layer of the classifier. The output of this layer is a n_c -dimensional vector, each component i of which denotes the probability that the data belongs to class i , for $i \in \{1, 2, \dots, n_c\}$ (please note that $n_c = 4$ in this example). The entropy is then given as

$$\mathcal{H}(X) = - \sum_{i=1}^{n_c} \hat{y}_i \log(\hat{y}_i) \quad (10)$$

where \hat{y} is the classifier's prediction for input X . In this figure, the points that are higher in color intensity are associated with a higher prediction uncertainty (i.e., entropy). As it is evident in the figure, the central purple cluster for CCLP and RESAD consists of the most uncertain data instances in the testing set. This is a significant finding and benefit of enforcing the CCLP loss: the method automatically forms a cluster, where the data instances that are hard to classify will be compactly clustered together. On the other hand, data instances that are easier to classify are compactly clustered away from this central cluster and into their own class-specific cluster. This important formation of the uncertain cluster is completely lost in the AE+C approach (top panel, right column), and we can see that hard-to-classify data instances are spread throughout the entire latent feature space.

One major benefit of the formation of such uncertain cluster is that we can design an active learning strategy to automatically identify the most informative subset of the unlabeled set. This set can be further reviewed and properly labeled by the SMEs and is part of our future work. This aspect emphasizes an important superiority of RESAD compared to CCLP, which is the size of the central purple cluster. This represents the number of data instances that the classifier has a high uncertainty about. In the case of CCLP approach, 21.87% of the data in the testing set are mapped to the central purple cluster, while the number for RESAD is 9.34%. This means that not only does our approach force the classifier to make more confident predictions on the unlabeled set (and a more accurate prediction according to Figure 3), but also reduces the size of the uncertain cluster significantly (1350 versus 570 data based on the size of the testing set). This means that SMEs have fewer data to review and label. Given how expensive and time-consuming the review process of each data is, this will result in significant savings in SMEs time and the data labeling cost.

In Figure 5, we further quantify the purity of the class-specific clusters by calculating the entropy of the class-distribution of the data instances that are mapped to each one of the class-specific clusters. Lower values mean that the class-specific clusters are purer and contain fewer data from other classes in them; we visualize the average value across the $n_c = 4$ class-specific clusters in the figure. Both RESAD and CCLP improve the purity of the clusters as more labeled data is provided, which is an intuitive result. AE+C, however, does not improve the purity of the clusters at all. This drawback of the AE+C approach is also evident in Figure 4, where the class-specific clusters are not compact and distant from one another.

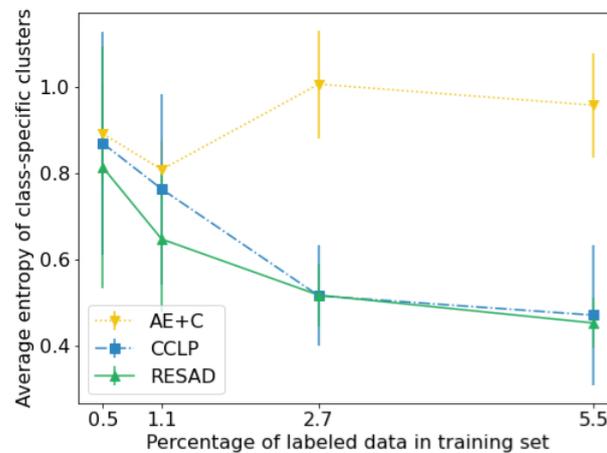


Figure 5. This figure shows the purity of the class-specific clusters in the latent space. The lower values indicate purer clusters.

4.5. Robustness to the Adversarial Perturbation

In this section, we investigate the robustness of our proposed approach, as well as the baseline methods against small but misleading noise (i.e., adversarial perturbations). This is an important experimentation that shows the reliability of model predictions in the presence of unwanted noise and is a crucial factor for operational models. We do this by implementing a perturbation scheme called the fast gradient sign method (FGSM) [42]. FGSM is a white-box adversarial perturbation method and generates adversarial examples in the presence of model parameters [43]. This perturbation scheme hypothesizes that neural networks are designed in a linear fashion (i.e., the components of neural networks, such as dot product, convolution, etc. are linear), and are vulnerable to linear adversarial noise. Such linear perturbation can be derived from

$$\tilde{X} = X + \epsilon \text{sign}(\nabla_x J(\eta, X, y)) \quad (11)$$

where ϵ is the magnitude of error, $\text{sign}(\cdot)$ is the sign function, η is the set of model parameters that affect the classification of X , i.e., $\eta = \{\phi, \psi\}$ (since input data X is first mapped to the latent feature space with the encoder, $q_\phi(z | x)$, and then is classified by the classifier, $c_\psi(y | z)$), and $J(\eta, X, y)$ is the loss function used for model training, which is depicted in Equation (1). Based on Equation (11), the adversarial noise is obtained by applying the sign function to the gradient of the loss function with respect to the input data. Based on our robustness evaluations of adversarial examples using the FGSM, Figure 6 shows that RESAD consistently and significantly outperforms the baseline CCLP and AE+C models for different percentages of perturbation. CCLP takes the second rank in average classification accuracy, and AE+C is the worst performing model. This figure shows the results for the case of a 1000-sample labeled set. However, the superiority of our approach holds over baselines with smaller labeled sets as well (Figure A3 in Appendix C).

We also report the effect of adversarial perturbation on the per-class F1-score of classification in Figure A7 in Appendix C. Please note that F1-score is the harmonic mean of precision and recall and is defined as follows,

$$\text{F1-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (12)$$

As it is evident, RESAD's superiority over the baseline methods is consistent across class-specific performance metrics (i.e., F1-score). CCLP, on the other hand, performs better than AE+C for majority classes (nominal and speed anomaly) and worse for minority classes (path anomaly and control anomaly).

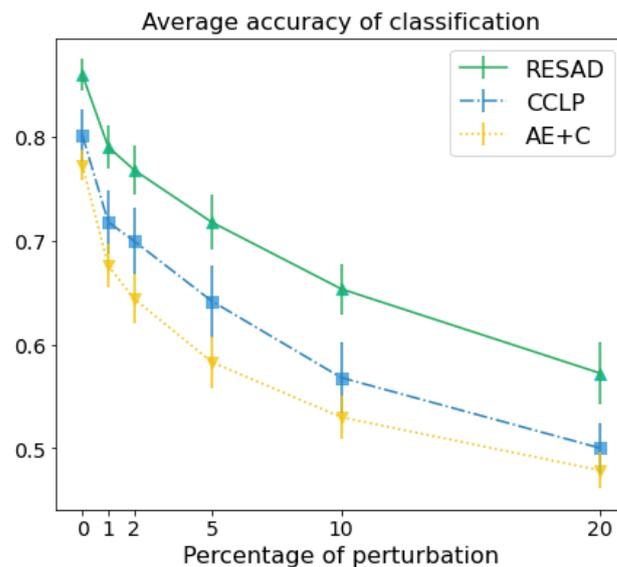


Figure 6. Comparison of average accuracy of classification for our proposed model, CCLP, and AE+C for different perturbation percentages.

In order to further improve the robustness of RESAD, we augmented the training optimization objective in Equation (1) based on two innovative ideas posed in the machine learning community recently: autoencoding variational autoencoder (VAE) [44] and interpolation consistency training (ICT) [45]. Both of these approaches have been developed to improve the consistency and robustness of the mapping from the input data to the latent feature space for down-stream tasks such as the one here (i.e., multi-class classification). Details of these approaches are depicted in Appendix B. However, we observed in Figure A3 (in Appendix B) that none of these augmentations result in any improvement in the robustness of RESAD to adversarial perturbation. We actually show that, in the case of a 1000-sample labeled set, AVAE and RESAD perform significantly close to each other and outperform ICT, while, in the case of a 200-sample labeled set, RESAD dominates both AVAE and ICT in the average accuracy of classification with different percentages of perturbation.

5. Conclusions

We proposed RESAD, a Robust and Explainable Semi-supervised learning model for Anomaly Detection in aviation data. Our proposed model is novel in several aspects, as follows: (1) It is semi-supervised: it addresses the shortcomings of supervised and unsupervised models in the aviation literature by taking into account both the majority unlabeled data and the minority labeled data sets. (2) It is explainable: the model incorporates graph-theoretic methods to propagate labels from the labeled set to the unlabeled set and form a compact structured feature space. This improves the interpretability of the learned latent feature space, where more information can be extracted for down-stream tasks such as active learning. (3) Lastly, it is robust to adversarial perturbations that significantly improves its reliability and applicability in the domain.

We evaluated the classification performance of RESAD against three existing methods in the literature. For this purpose, we developed a real-world case study of multi-class anomaly detection using commercial aircraft flight data during approach to landings. First, we illustrated the superiority of the semi-supervised learning over a supervised method in the aviation literature (Figure 3) when the size of labeled data is small. We specifically showed that, with 5.5% of training data labeled, the supervised model (DT-MIL) finds anomalies with 54.2% accuracy, while the semi-supervised models are significantly more accurate with 77.2% (AE+C), 80.1% (CCLP), and 86% (RESAD) average accuracy.

We further quantified the interpretability of the learned latent feature space by the three semi-supervised models. We show qualitatively (Figure 4) and quantitatively (Figure 5) that methods which induce supervision into their feature learning and encoding (CCLP and RESAD) build an interpretable latent feature space. This well-structured latent space is advantageous because it explains which regions in this space are compactly populated by each of the labeled anomaly classes. This is observed when clusters with high class purity were formed using unsupervised clustering and corroborated with the t-SNE visualization. This important trait allows for intelligent sampling from each region to select the most informative data for future labeling efforts (i.e., active learning). On the other hand, the AE+C approach learns a latent feature space that is not compact and representative for more advanced down-stream tasks. This is due to the fact that AE+C does not induce any supervision in the feature learning. Moreover, the purity of the class-specific clusters shaped in the latent feature space does not improve in the AE+C method with an increase in the size of the labeled set (Figure 5).

Lastly, we quantified the robustness of the three semi-supervised methods against adversarial perturbations induced in the input data space and show that RESAD significantly outperforms CCLP and AE+C (Figure 6). We specifically show that with a relatively high level of adversarial perturbation at 10% (according to Equation (7)), RESAD's performance is at 65.3% average accuracy (a drop of 20.6 percentage point (pp) compared to no perturbation), while CCLP and AE+C performances are at 56.8% and 53% average accuracy, respectively (drop of 23.3pp for CCLP and 24.2pp for AE+C). We further compared the robustness of RESAD against augmentations based on two recent studies in machine learning literature (Figure A3), and showed that none of those augmentations result in an improvement in the robustness of our model and it results in a loss of performance (ICT for both smaller and larger labeled sets and AVAE for smaller labeled set).

Potential future directions: One potential direction of future work is to extend the semi-supervised model to an open-set recognition model. In a testing scenario, this model would be capable of rejecting a new data as belonging to any of known classes and labeling them as unknown. This would be an important step forward in detecting unknown vulnerabilities and anomalies. Different metrics obtained by the model such as the reconstruction error, entropy of the classifier's prediction, and/or distance to the centroid of the assigned cluster in the latent feature space can be used to develop such capability. Another more practical extension of the model is to examine methods that shed light on the inference made by the model such as integrated gradients [46] or SHAP values [47]. These methods propagate back the output of the model's classifier to the input space to identify what features at what specific time window were influential in the model's decision making. These explanations from the original input space can help with model validation and promote acceptance within the domain.

Author Contributions: M.M. and B.M. conceived the idea together. B.M. provided domain expertise and set up the FOQA case study. M.M. developed the methodology and implementations of the method and performed validations and obtained the results. A.A.A. developed the comprehensive literature review and implemented the validation of robustness to perturbation. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the NASA Airspace Operation and Safety Program and the NASA System-wide Safety Project.

Data Availability Statement: The raw data that is used to obtain the results in this paper is available at <https://c3.nasa.gov/dashlink/projects/85/>, accessed on 1 August 2022. The processed version of the data is currently under review and will be shared at the same link as above.

Acknowledgments: Authors acknowledge the funding of this research from NASA System-wide Safety Project under contracts 80ARC020D0010 and NNA16BD14C.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

SME	Subject Matter Expert
VAE	Variational Auto-Encoder
FOQA	Flight Operational Quality Assurance
NTSB	National Transportation Safety Board
DT-MIL	Deep Temporal - Multiple Instance Learning
BCE	Binary Cross Entropy
MSE	Mean Squared Error
CCLP	Compact Clustering via Label Propagation
AE+C	Auto-Encoder + Classifier
t-SNE	t-Stochastic Neighbor Embedding
ICT	Interpolation Consistency Training

Appendix A. Model Architecture

Figure A1 shows the exact architecture of the encoder. The input data goes through three parallel branches of 1D convolution operation with different filter sizes (the first numeric) and kernel sizes (the second numeric) followed by batch normalization (BN) and ReLU activation function, and finally a max pooling with size 2. The decoder is identical to the encoder by just swapping the 1D convolution with 1D transpose convolution and the max pooling with the up-sampling.

Figure A2 shows the architecture of the classifier. It consists two fully connected layers with 100 neurons each and ReLU activation function, and a dropout with 50% rate in between. The output of the second layer goes into a linear layer with Softmax activation and n_c number of neurons, where n_c is the number of classes in the training data.

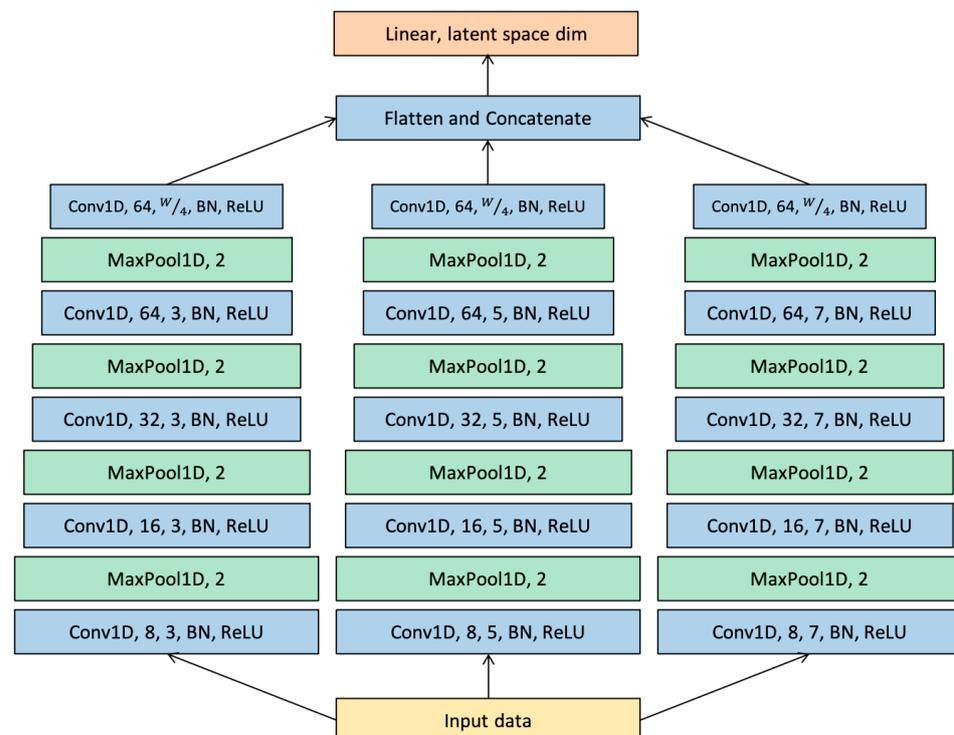


Figure A1. Exact architecture of the encoder.

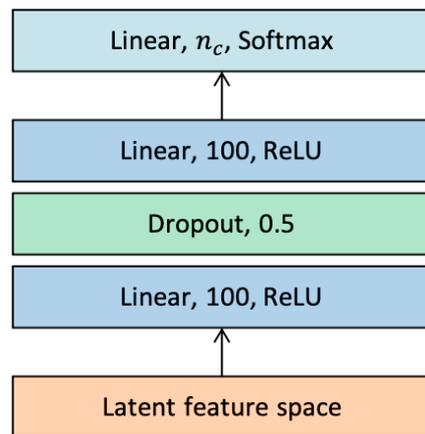


Figure A2. Exact architecture of the classifier.

Appendix B. Methods to Improve Robustness

In order to further improve the robustness of RESAD, we augmented the training optimization objective in Equation (1) based on two innovative ideas posed in the machine learning community recently, named Autoencoding VAE (AVAE) [44] and interpolation consistency training (ICT) [45]. Here, we first shortly summarize these approaches and then compare their performance with our proposed model.

Appendix B.1. Autoencoding Variational Auto-Encoder-AVAE

One of the approaches that was recently proposed to improve the consistency of the encoding obtained by the VAEs is AVAE [44]. The overall idea is to improve the consistency and robustness to perturbation that is lacking in the VAEs. In order to do that, Cemgil et al. [44] defines a reconstruction of the original data input X as a delusion (or auxiliary observation) \tilde{X} and the encoding of the delusion in the latent space as Z' . Then, in the optimization objective of training, they add an extra term to maximize the correlation of the encoding of the input data (i.e., Z), and the encoding of the delusion (i.e., Z'). The main idea is that by enforcing a high correlation between these two encodings (one coming from original data and one from a sample of its reconstruction), the VAE would be more consistent in the encoding and more robust to adversarial perturbations. They evaluate the effectiveness of this approach compared to VAE based on supervised down-stream classification tasks in the latent feature space.

Appendix B.2. Interpolation Consistency Training-ICT

ICT is a semi-supervised classification approach that improves performances by moving the decision boundaries to regions with low data density [45]. Such configuration is achieved through encouraging a prediction of interpolated unlabeled data to be in harmony with the interpolation of corresponding predictions. The interpolation step is computed by

$$\text{Mix}_\lambda(a, b) = \lambda \cdot a + (1 - \lambda) \cdot b \quad (\text{A1})$$

where λ is the interpolation coefficient, and a and b are the inputs. Using Equation (A1), we can define a prediction model f_ζ that provides similar predictions at interpolations of unlabeled data:

$$f_\zeta\left(\text{Mix}_\lambda\left(X_U^i, X_U^j\right)\right) \approx \text{Mix}_\lambda\left(f_{\zeta'}\left(X_U^i\right), f_{\zeta'}\left(X_U^j\right)\right) \quad (\text{A2})$$

where ζ' is a moving average of ζ . The study uses a mean teacher–student architecture to train a model on the unlabeled data and, in parallel, uses the limited labeled data to train the classifier f_ζ . The overall model is trained based on minimizing the classification objective combined with a weighted consistency loss. In the original paper [45], ICT outperformed supervised and mean teacher methods, demonstrating effectiveness in delineating optimum boundaries.

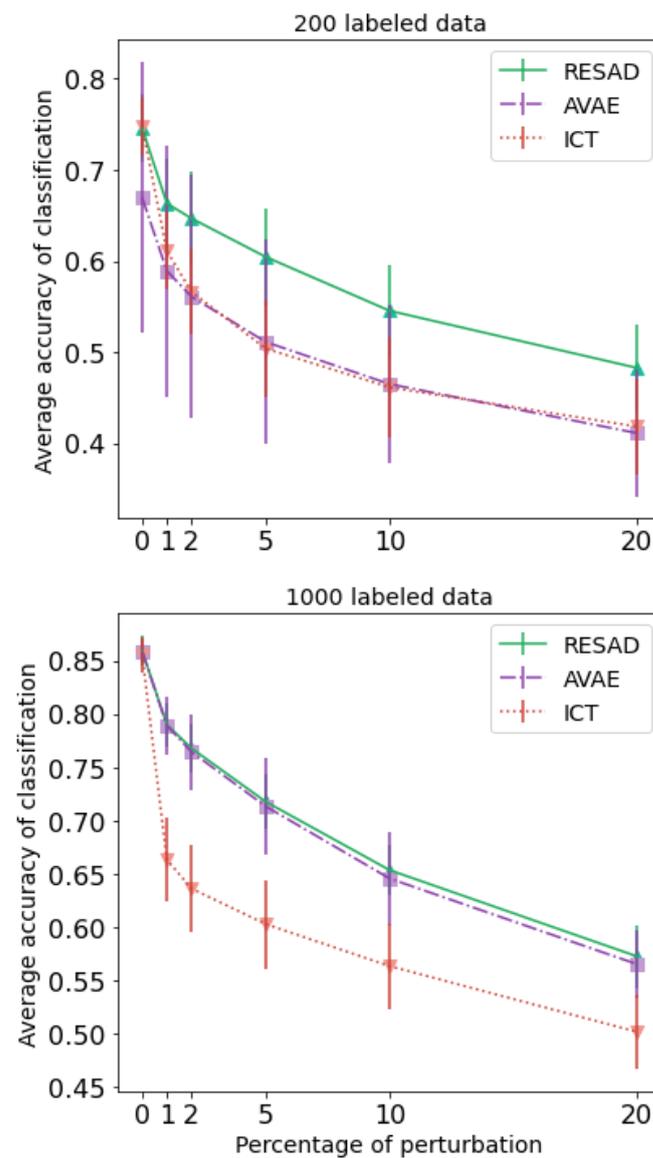


Figure A3. Average classification accuracy comparison for our proposed model compared to AVAE and ICT, with 200-sample and 1000-sample labeled data, with respect to increasing perturbation percentages.

Appendix C. Additional Tables and Figures

Table A1. Average performance of the models on the binarized anomaly detection problem (numbers show mean (standard deviation) based on 20 independent trials of training). Bold text shows the best performance in each column.

Method	Precision	Recall	F1-Score	AUROC
100 (0.55 %) labeled data				
DT-MIL	0.38(0.02)	0.80(0.08)	0.51(0.02)	0.62(0.05)
AE+C	0.52(0.03)	0.86(0.03)	0.64(0.02)	0.81(0.02)
CCLP	0.59(0.07)	0.71(0.14)	0.63(0.07)	0.76(0.06)
RESAD	0.63(0.04)	0.71(0.12)	0.66(0.05)	0.79(0.04)
200 (1.1 %) labeled data				
DT-MIL	0.41(0.05)	0.77(0.1)	0.53(0.03)	0.66(0.03)
AE+C	0.55(0.03)	0.89(0.02)	0.68(0.02)	0.85(0.01)
CCLP	0.60(0.03)	0.84(0.04)	0.70(0.02)	0.83(0.02)
RESAD	0.64(0.04)	0.84(0.03)	0.72(0.03)	0.85(0.02)

Table A1. *Cont.*

Method	Precision	Recall	F1-Score	AUROC
500 (2.7 %) labeled data				
DT-MIL	0.46(0.05)	0.74(0.07)	0.57(0.04)	0.72(0.04)
AE+C	0.61(0.02)	0.92(0.03)	0.73(0.02)	0.91(0.01)
CCLP	0.66(0.03)	0.92(0.03)	0.77(0.02)	0.90(0.01)
RESAD	0.70(0.04)	0.90(0.05)	0.78(0.03)	0.88(0.03)
1000 (5.5 %) labeled data				
DT-MIL	0.49(0.05)	0.78(0.07)	0.60(0.02)	0.76(0.02)
AE+C	0.64(0.02)	0.94(0.01)	0.76(0.01)	0.93(0.01)
CCLP	0.67(0.03)	0.95(0.01)	0.78(0.02)	0.92(0.01)
RESAD	0.75(0.02)	0.94(0.01)	0.83(0.01)	0.93(0.01)

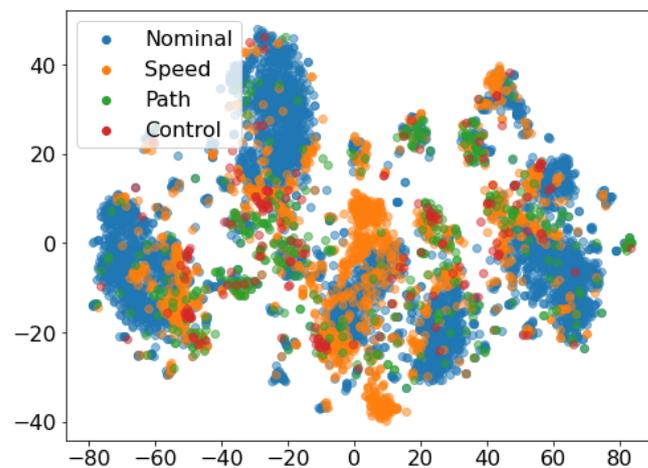


Figure A4. This figure shows the 2D visualization of the flight time-series using t-SNE, color-coded based on the true class to which each data belongs.

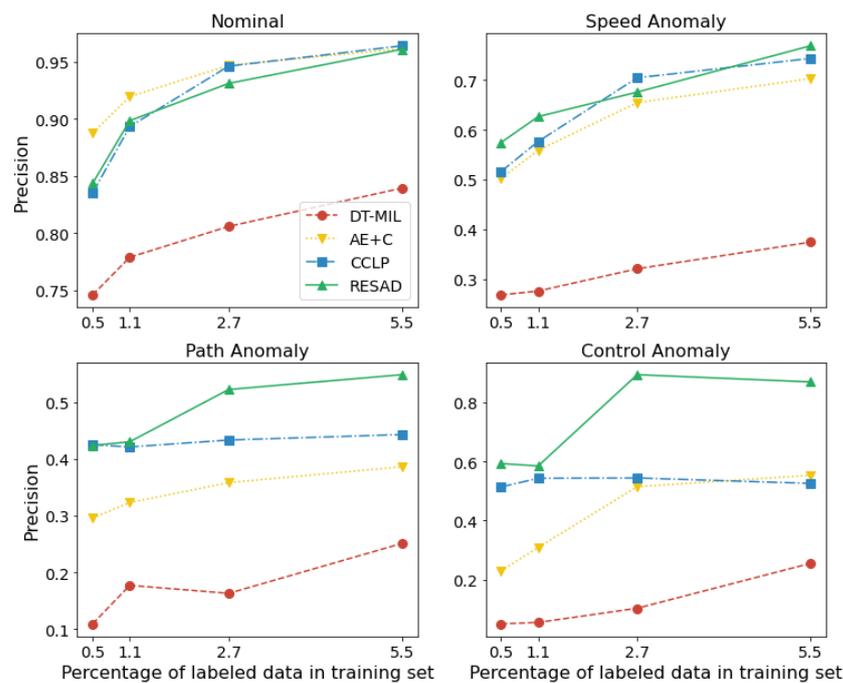


Figure A5. This figure compares performance (precision per class) of our proposed model (colored green) with multiple baseline methods, such as DT-MIL (red), AE+C (yellow), and CCLP (blue).

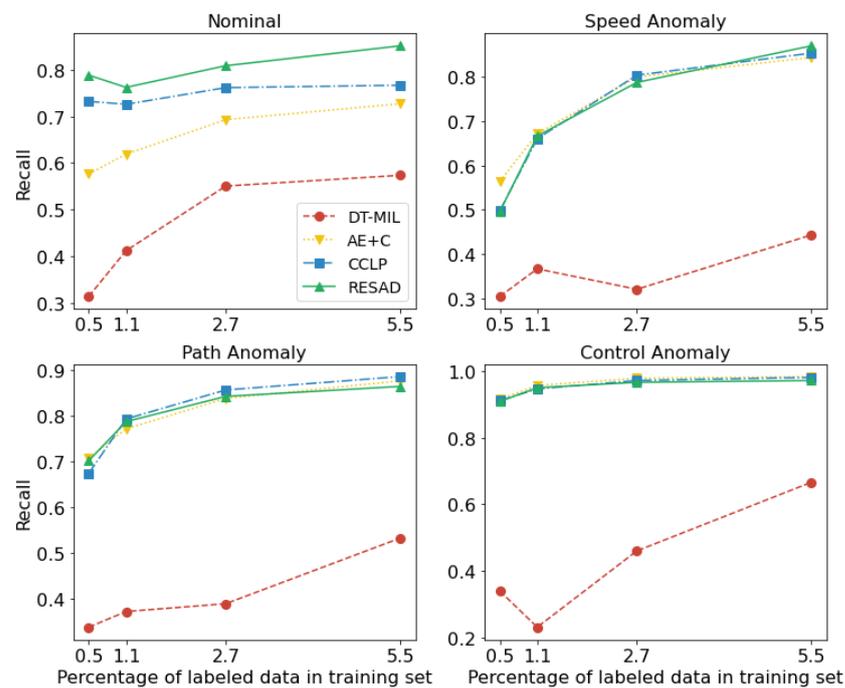


Figure A6. This figure compares performance (recall per class) of our proposed model (colored green) with multiple baseline methods, such as DT-MIL (red), AE+C (yellow), and CCLP (blue).

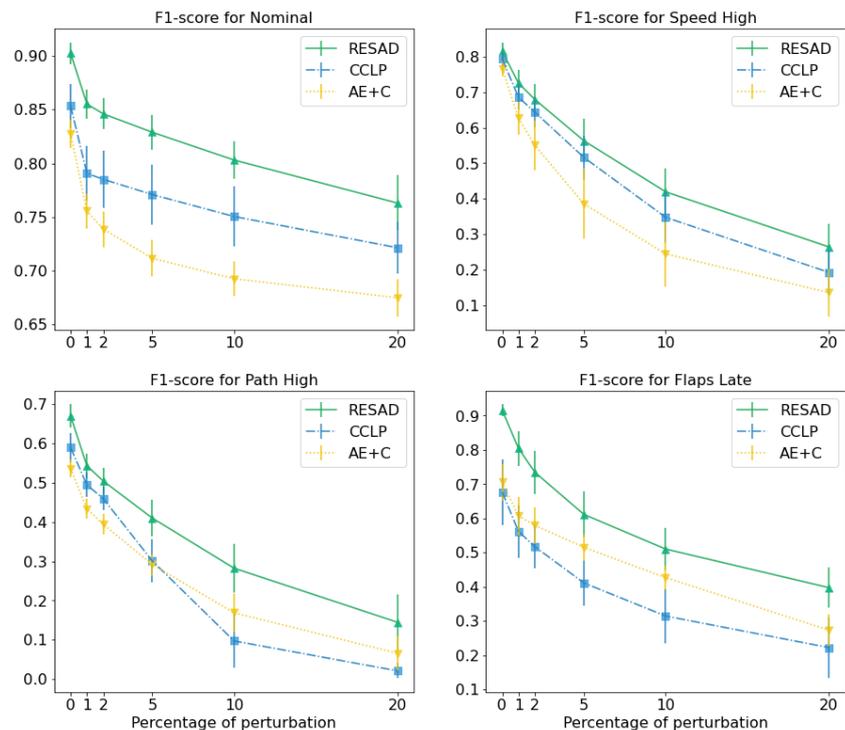


Figure A7. Comparison of our proposed model, CCLP, and AE+C for per-class F1-score with respect to increasing percentages of adversarial perturbation.

References

1. Howe, J. The Rise of Crowdsourcing. *Wired Mag.* **2006**, *14*, 1–4.
2. Kimberly, M.S.C. Coral reef video game will help create global database. *EOS* **2018**, *99*. [[CrossRef](#)]
3. Jordan Raddick, M.; Georgia Bracey, P.L.G. Exploring the Motivations of Citizen Science Volunteers. *Jan Vandenberg Astron. Educ. Rev.* **2010**, *9*, 1–4. [[CrossRef](#)]

4. Kamalika D.; Nikunj, C.; Oza, B.M. *Ask-the-Expert: Minimizing Human Review for Big Data Analytics through Active Learning*; NASA/TM—2019–220337; NASA Langley Research Center: Hampton, VA, USA, 2019.
5. Chapelle, O.; Scholkopf, B.; Zien, A. *Semi-Supervised Learning*; The MIT Press: Cambridge, MA, USA, 2006.
6. *Airplane Flying Handbook*; Federal Aviation Administration: Newcastle, WA, USA, 2016.
7. Janakiraman, V.M.; Nielsen, D. Anomaly detection in aviation data using extreme learning machines. In Proceedings of the 2016 International Joint Conference on Neural Networks, IJCNN 2016, Vancouver, BC, Canada, 24–29 July 2016; pp. 1993–2000.
8. Puranik, T.G.; Rodriguez, N.; Mavris, D.N. Towards nline prediction of safety-critical landing metrics in aviation using supervised machine learning. *Transp. Res. Part Emerg. Technol.* **2020**, *120*, 102819. [[CrossRef](#)]
9. Das, S.; Matthews, B.L.; Srivastava, A.N.; Oza, N.C. Multiple kernel learning for heterogeneous anomaly detection: Algorithm and aviation safety case study. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 25–28 July 2010; pp. 47–56.
10. Das, S.; Matthews, B.L.; Lawrence, R. Fleet level anomaly detection of aviation safety data. In Proceedings of the IEEE Conference on Prognostics and Health Management, Denver, CO, USA, 20–23 June 2011; pp. 1–10.
11. Basora, L.; Olive, X.; Dubot, T. Recent advances in anomaly detection methods applied to aviation. *Aerospace* **2019**, *6*, 117. [[CrossRef](#)]
12. Bay, S.D.; Schwabacher, M. Mining distance-based outliers in near linear time with randomization and a simple pruning rule. In Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 24–27 August 2003; pp. 29–38. [[CrossRef](#)]
13. Iverson, D.L. Inductive System Health Monitoring. In Proceedings of the International Conference on Artificial Intelligence, Nevada, FL, USA, 21–24 June 2004.
14. Matthews, B.; Srivastava, A.N.; Schade, J.; Schleicher, D.; Chan, K.; Gutterud, R.; Kiniry, M. Discovery of Abnormal Flight Patterns in Flight Track Data. In Proceedings of the 2013 Aviation Technology, Integration, and Operations Conference, Los Angeles, CA, USA, 12–14 August 2013. [[CrossRef](#)]
15. Li, L.; Das, S.; HAnsmann, R.; Palacios, R.; Srivastava, A. Analysis of Flight Data Using Clustering Techniques for Detecting Abnormal Operations. *J. Aerosp. Inf. Syst.* **2015**, *12*, 587–598. [[CrossRef](#)]
16. Memarzadeh, M.; Matthews, B.; Avrek, I. Unsupervised Anomaly Detection in Flight Data Using Convolutional Variational Auto-Encoder. *Aerospace* **2020**, *7*, 115. [[CrossRef](#)]
17. Hinton, G.; Salakhutdinov, R. Reducing the Dimensionality of Data with Neural Networks. *Science* **2006**, *313*, 504–507. [[CrossRef](#)]
18. Kingma, D.; Welling, M. Auto-Encoding Variational Bayes. In Proceedings of the International Conference on Learning Representations (ICLR), Scottsdale, AZ, USA, 2–4 May 2013.
19. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *arXiv* **2014**, arXiv:1406.2661 [[CrossRef](#)]
20. An, J.; Cho, S. *Variational Autoencoder Based Anomaly Detection Using Reconstruction Probability*; Technical Report; SNU Data Mining Center: Seoul, Korea, 2015.
21. Xu, H.; Wenxiao, C.; Zhao, N.; Li, Z.; Bu, J.; Li, Z.; Liu, Y.; Zhao, Y.; Pei, D.; Feng, Y.; et al. Unsupervised Anomaly Detection via Variational Auto-Encoder for Seasonal KPIs in Web Applications. In Proceedings of the 2018 World Wide Web Conference, Lyon, France, 23–27 April 2018; pp. 187–196.
22. Chen, R.Q.; Shi, G.H.; Zhao, W.L.; Liang, C.H. Sequential VAE-LSTM for Anomaly Detection on Time Series. *arXiv* **2019**, arXiv:1910.03818
23. Zhang, C.; Chen, Y. Time Series Anomaly Detection with Variational Autoencoders. *arXiv* **2019**, arXiv:1907.01702.
24. Park, D.; Hoshi, Y.; Kemp, C.C. A Multimodal Anomaly Detector for Robot-Assisted Feeding Using an LSTM-Based Variational Autoencoder. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1544–1551. [[CrossRef](#)]
25. Su, Y.; Zhao, Y.; Niu, C.; Liu, R.; Sun, W.; Pei, D. Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '19, Anchorage, AK, USA, 4–8 August 2019; pp. 2828–2837.
26. Wang, L.; Lucic, P.; Campbell, K.; Wanke, C. Autoencoding Features for Aviation Machine Learning Problems. *arXiv* **2020**, arXiv:2011.01464.
27. Reddy, K.K.; Sarkar, S.; Venugopalan, V.; Giering, M. Anomaly Detection and Fault Disambiguation in Large Flight Data: A Multi-modal Deep Auto-encoder Approach. In Proceedings of the Annual Conference of the Prognostics and Health Monitoring Society, Denver, CO, USA, 2–8 October 2016; p. 7.
28. Lee, H.; Madar, S.; Sairam, S.; Puranik, T.G.; Payan, A.P.; Kirby, M.; Pinon, O.J.; Mavris, D.N. Critical parameter identification for safety events in commercial aviation using machine learning. *Aerospace* **2020**, *7*, 73. [[CrossRef](#)]
29. Janakiraman, V.M. Explaining Aviation Safety Incidents Using Deep Temporal Multiple Instance Learning. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, London, UK, 19–23 August 2018; pp. 406–415. [[CrossRef](#)]
30. Sharma, M.; Das, K.; Bilgic, M.; Matthews, B.; Nielsen, D.; Oza, N. Active learning with rationales for identifying operationally significant anomalies in aviation. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Garda, Italy, 19–23 September 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 209–225.

31. Das, K.; Avrek, I.; Matthews, B.; Sharma, M.; Oza, N. Ask-the-expert: Active learning based knowledge discovery using the expert. In Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery (ECML-PKDD), Skopje, Macedonia, 18–22 September 2017.
32. Sahasrabhojane, A.; Iverson, D.; Wolfe, S.; Bradner, K.; Oza, N. Active Learning Strategies to Reduce Anomaly Detection False Alarm Rates. In Proceedings of the 37th International Conference on Machine Learning, PMLR108, Online, 13–18 July 2020.
33. Memarzadeh, M.; Matthews, B.; Templin, T.; Sharif Rohani, A.; Weckler, D. Novel Active Learning Framework for Anomaly Detection in Aviation with Expert in the Loop. In Proceedings of the AIAA SCITECH 2022 Forum, San Diego, CA, USA, 3–7 January 2022. [[CrossRef](#)]
34. Yan, W. Detecting gas turbine combustor anomalies using semi-supervised anomaly detection with deep representation learning. *Cogn. Comput.* **2020**, *12*, 398–411. [[CrossRef](#)]
35. Jiang, J.R.; Kao, J.B.; Li, Y.L. Semi-supervised time series anomaly detection based on statistics and deep learning. *Appl. Sci.* **2021**, *11*, 6698. [[CrossRef](#)]
36. Memarzadeh, M.; Matthews, B.; Templin, T. Multiclass Anomaly Detection in Flight Data Using Semi-Supervised Explainable Deep Learning Model. *J. Aerosp. Inf. Syst.* **2022**, *19*, 83–97. [[CrossRef](#)]
37. Kingma, D.; Rezende, D.; Mohamed, S.; Welling, M. Semi-supervised learning with deep generative models. *Adv. Neural Inf. Process. Syst.* **2014**, *27*. Available online: <https://papers.nips.cc/paper/2014/hash/d523773c6b194f37b938d340d5d02232-Abstract.html> (accessed on 1 August 2022).
38. Kamnitsas, K.; Castro, D.; Le-Folgoc, L.; Walker, I.; Tanno, R.; Rueckert, D.; Glocker, B.; Criminisi, A.; Nori, A. Semi-supervised learning via compact latent space clustering. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm Sweden, 10–15 July 2018; pp. 2459–2468.
39. Joachims, T. Transductive inference for text classification using support vector machines. In Proceedings of the Sixteenth ICML, San Francisco, CA, USA, 27 June 1999; pp. 200–209.
40. van der Maaten, L.; Hinton, G. Visualizing Data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
41. Kingma, D.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations (ICLR), Banff, AB, Canada, 14–16 April 2014.
42. Goodfellow, I.; Shlens, J.; Szegedy, C. Explaining and harnessing adversarial examples. *arXiv* **2014**, arXiv:1412.6572.
43. Ebrahimi, J.; Rao, A.; Lowd, D.; Dou, D. Hotflip: White-box adversarial examples for text classification. *arXiv* **2017**, arXiv:1712.06751.
44. Cemgil, T.; Ghaisas, S.; Dvijotham, K.; Gowal, S.; Kohli, P. The Autoencoding Variational Autoencoder. *Adv. Neural Inf. Process. Syst. (NeurIPS)* **2020**, *33*, 15077–15087.
45. Verma, V.; Lamb, A.; Kannala, J.; Bengio, Y.; Lopez-Paz, D. Interpolation Consistency Training for Semi-supervised Learning. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI), Yokohama, Japan, 7–15 January 2021; pp. 3635–3641. [[CrossRef](#)]
46. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic Attribution for Deep Networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, NSW, Australia, 6–11 August 2017; pp. 3319–3328.
47. Lundberg, S.; Lee, S. A Unified Approach to Interpreting Model Predictions. *arXiv* **2017**, arXiv:1705.07874.