

Deep Contrastive Anomaly Detection for Airline Ancillaries Prediction

Pu Yang

Department of Electrical and Electronic Engineering
Imperial College London
London, UK
py715@ic.ac.uk

Arinbjörn Kolbeinsson

Department of Epidemiology and Biostatistics
Imperial College London
London, UK
ak711@ic.ac.uk

Naman Shukla

Deepair Solutions
London, UK
naman@deepair.io

J. A. Barria*

Department of Electrical and Electronic Engineering
Imperial College London
London, UK
jab@ic.ac.uk

Abstract—The increasing range of ancillary products offered by airlines is making existing, static frameworks obsolete. The changing expectations of customers have created a need for more dynamic and reactive offers. In order to tailor an offer to an individual journey, it is possible to leverage similar journeys and the observed outcomes in a semi-supervised approach. In this paper, a multi-stage deep learning framework, namely Deep Ancillaries Prediction (DAP), is developed to understand personalised demand for airline ancillaries and improve pricing strategies. DAP aims to solve the overlapping distribution problem and class imbalances observed in real-world airline datasets. The framework incorporates a contrastive learning module to learn richer feature embeddings and an autoencoder for semi-supervised learning into one framework, and outperforms current ancillary prediction systems. The modules can be trained separately and hence, are suitable for an online learning setting. This framework is designed to be transferable to different prediction tasks in the airline industry. Significant performance enhancements are attained compared to the current state-of-the-art algorithms.

Index Terms—Airline ancillaries prediction, anomaly detection, contrastive learning, deep learning.

I. INTRODUCTION

The airline industry has rapidly un-bundled its products to offer multiple separate ancillaries, e.g., additional luggage, express check-in, or seat selection, to customers [1], [2]. This has dramatically changed the airlines' revenue stream. In 2019, the total revenue from ancillaries in the airline industry was over \$100 billion [3]. Separating out individual ancillaries has created a new challenge around what ancillaries should be offered or priorities for customers. Currently, offered ancillaries and their prices are typically pre-defined by the companies and cannot be adjusted dynamically based on customers' information provided, which limits both business revenues and customer satisfaction as the market and customers' preferences change over time. Therefore, it is necessary to design a framework that incorporates personalisation, trends and variations.

However, for such a framework to be built, multiple challenges need to be overcome.

The first challenge is the diversity and variance in customers. This is confounded by the fact that the statistical properties of customers who purchase and not-purchase an ancillary have high overlap. Secondly, the majority of the data comes from the not-purchased group, making the classification task highly imbalanced. This adds additional difficulty to the classification boundaries for these two groups of customers, as they are not easy to discern, making the decision process more complex. Therefore, traditional machine learning (ML) algorithms like support vector machine (SVM) [4] and random forest (RF) [5] can hardly be applied directly for such tasks. In this paper, the usefulness of Deep Learning (DL) techniques is investigated to explore the relationship between the customers' input and the actual purchase of an additional ancillary and hence, making the prediction based on their input. Such analysis can provide useful information to the business for updating the pricing policy for offered ancillaries as well as reveal the correlation between the willingness to pay for additional ancillaries and data provided by the customers.

In this paper, we propose a multi-stage ancillaries prediction framework, namely Deep Ancillaries Prediction (DAP), which is based on several DL techniques aiming to predict the most suitable ancillaries for customers. The feature extraction stage of DAP will train an extractor using contrastive learning (CL), which will learn a new feature embedding with closer sample distances inside each class and further distance between the classes. For the prediction stage, both the supervised feed-forward neural network (FFNN) and semi-supervised autoencoder (AE) models are used for the prediction. Note that the AE model is used to learn the hidden representation of the majority class, which can also provide its compressed features to other anomaly detection algorithms, such as One-class SVM [4], to improve the algorithm's performance. The training stage of the framework is detachable, where the feature extractor and the predictor can be updated separately; therefore, DAP

*Corresponding author

can be perfectly deployed in a real-world online setting. In DAP, we combine the advantages of producing distinct feature embedding from contrastive learning and anomaly detection techniques into one framework, aiming to improve the current ancillaries prediction system in the airline industry.

II. BACKGROUND

Recommender systems aim to provide an estimation of users' preferences on certain products [6], [7]. Three are the main categories of the current recommender models [8]: (i) Collaborative filtering models, which utilise the historical information from users and make a prediction based on other users with similar interests [9]; (ii) Content-based models, which analyse the correlation between the product and the auxiliary information from users [10]; (iii) Hybrid models, which combine different strategies to provide the recommendation [11]. Deep learning (DL) techniques, due to their ability to learn hidden representations, are being widely used in many fields, such as natural language processing and computer vision. In the recommender system, the usage of DL can help handle multimodal data, including textual data and image data, extract features from such data, and provide a more accurate end-to-end recommendation based on all the available data from the user [6], [8].

Contrastive learning (CL) aims at pulling similar samples closer while pushing others further away in a new feature space; it is commonly used in the computer vision field to classify image identities [12], [13]. The class of data that is being used as the reference for learning is called *Anchor*; during the training, several *Positive* pairs from the same class will be used to contrast against the *Negative* pairs from other classes. A CL-based self-supervised encoder with a latent space projection head is proposed in [12] while utilising the augmentation of the *Anchor* as additional *Positives* to help produce the new feature embedding. A supervised CL-based method proposed in [14] utilises label information to further enhance the image classification performance. Regarding the application of CL in anomaly detection, in [15], CL is used to learn a normal driving pattern, and the abnormal pattern will be detected when a certain contrastive based-threshold is met.

Anomaly detection is the process to find the data or patterns that are dissimilar to the regular data distribution or deviate from the entire dataset [16]. Detecting such deviated data points can provide early warnings of certain tragedies, such as monitoring heart attack using Electrocardiograms (ECG) signals, predicting network traffics, and preventing malicious activities in network security [17]–[20]. DL techniques have attracted more and more attention in the recent decade as it can solve problems end-to-end given the dataset. The autoencoder (AE), with its encoder and decoder structure, can learn hidden representations of the high-dimensional data in a lower dimension using the encoder unit and reconstruct the hidden features using the decoder. Its variant, the variational autoencoder (VAE), replaces the bottleneck of AE with parameters from a prior distribution, to prevent overfitting and enabling a generative process using the prior distribution [21]. In [22], an

AE-based network intrusion detection framework is proposed and able to handle high dimensionality datasets while capturing the non-linear correlation between the features using the hidden representations. Similar ideas proposed in [23], namely DAGMM, also use AE to extract the hidden representation of high-dimensional data and add additional reconstruction features such as Cosine and Euclidean Similarity to evaluate the reconstruction process; such features are further passed to a Gaussian Mixture Model (GMM) to predict the class probability in a semi-supervised way. In [24], a VAE-based recurrent neural network is proposed to monitor for anomalies during solar energy generation. Similar to DAGMM, the expectation vector from the VAE distribution can also be used as a compressed feature for other classification algorithms [25].

Pricing optimisation in the airline industry has previously revolved around improving revenue by changing the base ticket price [26], also known as the right-to-fly. This is changing as ancillaries become a more critical source of revenue. Another recent change is increased emphasis on customer satisfaction. The current state-of-the-art (SOTA) in airline ancillary prediction, proposed in [27], jointly optimises for revenue and conversion rate of offers as an indirect proxy for customer satisfaction. The method has been deployed in production and outperforms previous systems [28]. The approach is supervised and utilises Feed-Forward Neural Networks (FFNN) with Cross-Entropy (CE) loss to predict the class probabilities of each data sample. Note that there exist some challenges in that framework, which limit the performance of the entire system: (i) additional classification biases could have been introduced due to equal weights being assigned to both classes during training; (ii) feature selection was made manually by industry experts, which hinders full utilisation of the available information; (iii) the statistical properties of the two data classes are highly overlapping, supervised models often struggle in such scenarios; (iv) the model is trained in an offline setting, which might not be an optimal solution if the market violates by other factors such as the COVID-19 pandemic. Therefore, to overcome the aforementioned difficulties, the proposed DAP aims at providing a multi-stage solution for feature extraction, fast and stable model prediction and retraining.

We advocate that the reconstruction-based framework is suitable for most public datasets, where the similarity between the majority and minority classes is low. However, for the actual industrial dataset, such as the airline ancillary prediction task where the class differences are less obvious, the reconstruction-based methods cannot provide useful features for the class prediction. The proposed DAP consists of a CL-based feature extractor, which is used to produce a new feature embedding with a closer sample distance for the same class of data while pushing the classes further away from each other. The feature embedding (with more distinct features) will then be used by the prediction module to improve the anomaly detection performance.

III. THE DEEP ANCILLARIES PREDICTION FRAMEWORK

The recent solution suggested in [27] has several limitations that restrict the performance of the system. The multi-stage framework proposed here divides the process into data pre-processing, feature extraction, and anomaly detection stages. The feature extractor (FE) is trained by CL, which aims to produce a better feature representation for the prediction module. Regarding the class imbalance problem with high dimensionality, DAP utilises a semi-supervised AE to train on the majority class of data; the compressed features from AE is further used by other anomaly detection algorithms to reduce the algorithm complexity. Experiments show that using CL improves the anomaly detection performance and simplify the classifier training process; hence, it is perfect for deploying in the industries where the majority and minority classes of data are similar, e.g., airline ancillaries prediction. The detail workflow of the proposed DAP is shown in Fig. 1.

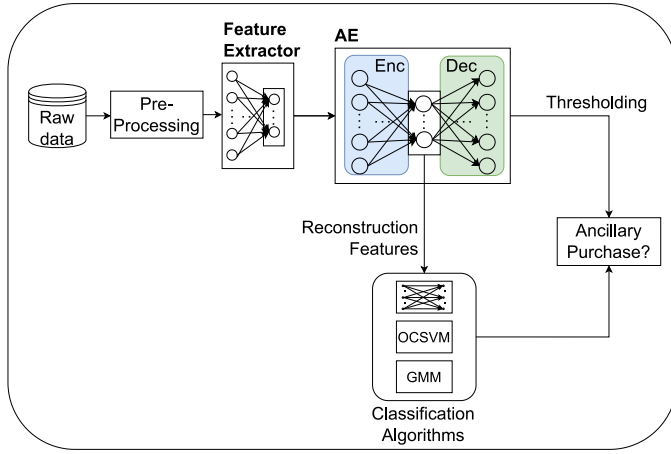


Fig. 1. The workflow of the proposed DAP Framework, which contains a data pre-processing stage, a feature extractor to learn a new feature embedding, and an AE-based anomaly detection module to learn the reconstruction process of the majority class, while its reconstruction features are used by downstream classification algorithms.

A. Data Pre-processing

The original dataset $\mathbf{D}$ we are using is provided by the international airline partner with *Deepair*, which contains the ancillary purchase information with 110 features lasting 18 months from April 2019 to September 2020, i.e., $\mathbf{D} \in \mathbb{R}^{53380 \times 110}$, $\mathbf{D} = \{x_1, \dots, x_{53380}\}$. Note that this period covers normal and abnormal markets across the COVID-19 pandemic, where the airline industry was hugely impacted; the proposed DAP is robust enough to handle such market changes and further results will be discussed in the following sections. Note also that the class imbalance problem exists in $\mathbf{D}$, which contains 82% of the class Ancillary-Not-Purchased (NP) and 18% of the class Ancillary-Purchased (P). In the anomaly detection perspective, the majority class, i.e., NP , is defined as the normal class, and the minority class, i.e., P becomes the anomaly class. Even though only the luggage ancillary is assessed in the paper, DAP is capable of any airline ancillaries

prediction since the data label can also be changed to express check-in, seat selection, or rental cars while the input features stay the same.

The data-processing stage includes data cleaning and encoding¹. The pre-processing stage contains three steps: (i) remove the redundant features that have no contribution to the task, i.e., the unique keys *session ID*, *request ID* in each record can be removed. (ii) convert the categorical features to one-hot encoding. For example, a total 48 airports were originally represented by a set of discrete indices $I = [0, 1, 2, \dots, 47]$ in a single column; the resulting encoded feature $\mathbf{X}_{\text{airport}} \in \mathbb{R}^{53380 \times 48}$. (iii) $\mathbf{D}$ contains future features that will not be accessible when the model is running, and such features have a direct relationship to the prediction labels. Examples like *ancillary quantity*, *payment success* will be removed to avoid dataset contamination. After these steps, the preprocessed dataset $\mathbf{X}$ will then be created to fully utilise the information provided by the source database, i.e., $\mathbf{X} \in \mathbb{R}^{53380 \times 477}$.

Figure 2 shows the correlation analysis of the features with respect to the class NP and P , where the difficulty of applying different ML algorithms is shown, as most of the features are not correlated to the label. Therefore, different DL techniques are being used to explore the non-linear relationships between the customers' input and their purchase decision. Moreover, a more informative feature embedding will be generated to enhance the detection performance, and potentially be adapted to other problems in the travel industry. This leads to the following section, Section III-B, where a contrastive learning-based FE is proposed.

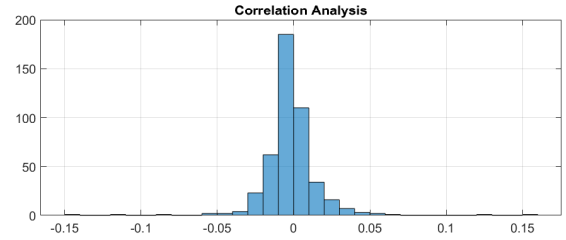


Fig. 2. Histogram plot for the pre-processed features, where most of the features have little correlation to the label.

B. Feature Extractor

The majority of the contrastive-learning methods, as discussed in Section II, target computer visions. In DAP, we extend CL-based methods to the airline industry for predicting the ancillaries in the anomaly detection perspective. Compared to the self-supervised way of CL, we choose supervised contrastive learning as the backbone of the FE to fully utilise the label information as the labels are easy to obtain without introducing additional work when collecting them. We note that by having the label as the support for CL, the range of *Positives* is extended to the same class of data. This reduces the effects of lacking data augmentation for the *Anchor* by

¹Detailed examples of the dataset cannot be provided due to confidentiality and data privacy.

providing enough *Positives* for the extractor to learn. This process is shown in Fig. 3, where *NP* is the majority class, *P* is the minority class. The FE proposed in this paper has an encoder $E(\cdot)$ to encode the original feature, and a projector $P(\cdot)$ following the discussion in [12], to map the encoded features to a latent space to enhance the learning process; it utilises the loss function proposed in [14] and is shown Equation 1. Note that the projection head will be dropped after the pre-training stage. A detailed illustration of the training process is depicted in Fig. 4.

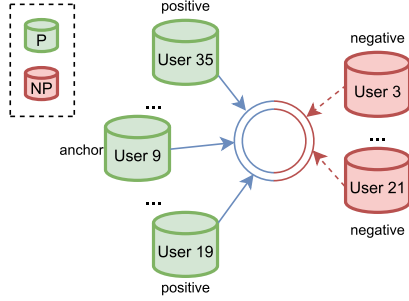


Fig. 3. Contrastive Learning to learn the feature embedding. *P* stands for the class Purchased, and *NP* is the class Not-Purchased. Data from these two classes are being contrasted during training, with a more separable feature embedding being generated.

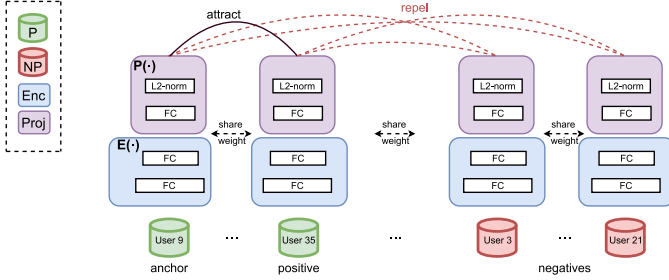


Fig. 4. The proposed feature extractor. $E(\cdot)$ encodes the original feature, $P(\cdot)$ projects and normalises the encoded embedding to a new latent space. The new embedding with more informative features from $E(\cdot)$ will be used by the predictor module in DAP.

$$\mathcal{L}^{CL} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(z_i \cdot z_p / \tau)}{\sum_{a \in A(i)} \exp(z_i \cdot z_a / \tau)} \quad (1)$$

Note that z is the output of the feature extractor, $i \in I = \{1 \dots n\}$, where n is the batch size, $P(i)$ is the set containing the same class of data with respect to anchor, $|P|$ is the cardinality of P , $A(i) = I \setminus \{i\}$, $a \setminus b$ is the removal operation to remove the element b from the set a , and τ is a scalar parameter to control the smoothness of the class boundary. Compared to the data augmentation of each image being used for CL in computer vision, DAP only utilises data from the same class, as sufficient data from each of the classes can be provided. In this way, we extend the application of CL from computer vision to the airline ancillaries prediction system in which the class features are similar and hard to be predicted. We

note that, although the bottleneck of AE also provides the compressed features, in this paper, the term *FE* refers only to the CL-based *Feature Extractor* discussed in this section.

C. Ancillaries Prediction

DAP is flexible and allows for supervised, semi-supervised and unsupervised prediction methods. For supervised learning, we implement a FFNN classifier to benchmark with the current solution proposed in [27]. For semi-supervised and unsupervised learning, several AE-based methods and general unsupervised methods are assessed. Note that we are focusing on the AE-based semi-supervised methods as most of the data comes from one category with high dimensionality.

1) *Supervised Neural Networks*: The supervised method involves a FFNN using fully connected layers (FC) with non-linear activation functions and the final layer of the Sigmoid function to predict the class probabilities. Regarding the loss function, binary cross-entropy loss (CE) is used as shown in Equation 2, where y is the class label and $\hat{y}_i$ is the predicted class probability.

$$\mathcal{L}^{CE}(y_i, \hat{y}_i) = \sum_{i=1}^n y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (2)$$

2) *Semi-supervised AE*: Several methods based on AE are implemented in DAP, focusing on the semi-supervised learning using the majority group of data for ancillary prediction: (i) AE with a loss-threshold strategy, where data with a higher loss value than a threshold δ will be classified as the anomaly class; (ii) using the compressed features from the bottleneck of AE as the input of other ML algorithms; and (iii) using the reconstruction features such as the similarity measurements proposed in [23], where a Gaussian Mixture Model (GMM) is co-trained with the AE to detect the anomalies.

An AE contains an encoder $\text{Enc}(\cdot)$ and a decoder $\text{Dec}(\cdot)$ such that $\mathbf{z} = \text{Enc}(\mathbf{x})$, and $\hat{\mathbf{x}} = \text{Dec}(\mathbf{z})$. Note that $\text{Enc}(\mathbf{x})$ projects $\mathbf{x}$ to a lower-dimensional subspace, i.e., $\mathbf{x} \in \mathbb{R}^{n \times d_x}$, $\mathbf{z} \in \mathbb{R}^{n \times d_z}$, and $d_z \ll d_x$, where n is the batch size, d is the dimensionality of the data. The objective function for AE is to minimise the reconstruction error $\mathcal{L}^{AE}$, where $\|\cdot\|$ is the L2-norm:

$$\mathcal{L}^{AE} = \frac{1}{n} \sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2 \quad (3)$$

Note that the error $\mathcal{L}^{AE}$ can be further used to formulate the classification rule C , which is defined as:

$$C(x_i) = \begin{cases} \text{normal}, & \text{if } \mathcal{L}^{AE} < \delta \\ \text{anomaly}, & \text{otherwise} \end{cases} \quad (4)$$

$\text{Enc}(\cdot)$ is a compression network (shown in Fig. 1-AE) for compressing the high-dimensional data $\mathbf{x} \in \mathbb{R}^{d_x}$ to $\mathbf{z} \in \mathbb{R}^{d_z}$, $d_z \ll d_x$. This is computed by using $\mathbf{z} = h_e(\mathbf{x}; \theta_e)$, where $h(\cdot)$ is the projection function with non-linear activation function, θ_e is the network parameters of the encoder. $\text{Dec}(\cdot)$

works similar to $\text{Enc}(\cdot)$, the only difference is that $\text{Dec}(\mathbf{z})$ reconstructs the lower dimensional feature $\mathbf{z}$ back to its original feature space. Note that additional reconstruction features can be added to $\mathbf{z}$ to improve the prediction performance when it is used in other ML algorithms: possible features include Cosine distance, Euclidean distance, and the reconstruction error shown in Equation 5, 6, and 7:

$$\text{Cos}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{\mathbf{x} \cdot \hat{\mathbf{x}}}{\|\mathbf{x}\| \|\hat{\mathbf{x}}\|} \quad (5)$$

$$\text{Euc}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{\|\mathbf{x} - \hat{\mathbf{x}}\|}{\|\mathbf{x}\|} \quad (6)$$

$$\text{Rec}(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2 \quad (7)$$

For example, let $\mathbf{z}_{\text{ae}} = \text{Enc}(\mathbf{x})$, $\mathbf{z}_s = \text{concat}(\text{Cos}; \text{Euc}; \text{Rec})$, the enhanced compressed feature $\mathbf{z} = \text{concat}(\mathbf{z}_{\text{ae}}; \mathbf{z}_s)$.

IV. RESULTS

In this section, the performance of DAP is evaluated. The empirical analysis includes the comparison between DAP and five strong existing approaches: (i) baseline supervised FFNN (SOTA from [27]); (ii) Semi-supervised AE using the rule C (Equation 4); (iii) general anomaly detection algorithms, i.e., One-Class SVM [4], Local Outlier Factors [29], Isolation Forest [30], and Deep SVDD [31]; (iv) the best algorithm in (iii) utilising the enhanced reconstruction features from AE; (v) Semi-supervised DAGMM [23]. Neural networks are built in Pytorch [32], while Scikit-learn is used for the general ML anomaly detection algorithms [33]. The initial learning rate we used is $5e^{-4}$, with the Adam optimiser being used to minimise the loss functions [34], and the batch size is set to 128. The newly pre-processed dataset $\mathbf{X}$ is split: 50% for training and 50% for evaluation. The training set is further divided 80:20 to create a training and validation set, where the validation set is used for model selection. The model output varies between approaches: class probabilities or scores from (i), (iii), and (iv); reconstruction errors from (ii); and sample energy from (v). The Receiver Operating Characteristic (ROC) can be calculated accordingly, and its corresponding Area Under the Curve (AUC) is used as the evaluation metric. Further evaluation metrics: Accuracy, Precision, Recall, and F1-score are also reported. Note that the experiments are repeated ten times, and the average of the metric with its standard deviation is reported. All the experiments are run using NVIDIA RTX 2080Ti GPU with INTEL I9-9900K CPU.

A. Feature Extractor

The feature extractor (FE) consists of an encoder $E(\cdot)$ and a projector $P(\cdot)$, it is designed as $P(E(\mathbf{x}))$ during the pre-training stage. Note that only $E(\cdot)$ will be used to produce the feature embedding for the prediction models. The encoder $E(\cdot)$ is structured as follows: Fully Connected layers (FC)(477, 512, ReLU)-FC(512, 256, ReLU)-FC(256,

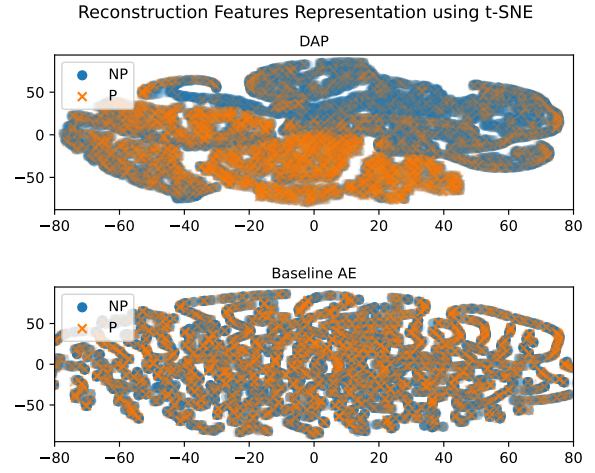


Fig. 5. Feature representation using t-SNE. The reconstruction features for the two classes are more separable in DAP (top) than in AE (bottom).

128, ReLU)-FC(128, 64, ReLU), and the projector $P(\cdot)$ has structure of FC(64, 32).

Fig. 5 visualises a 2-dimensional projection of the reconstruction features $\mathbf{z} = [\mathbf{z}_{\text{ae}}; \text{Cos}; \text{Rec}]$ using t-SNE [35] for DAP-AE and the baseline AE methods. For the top of the figure where DAP features are involved, the class P and NP are more separable than the baseline reconstruction features on the bottom. This also proves that the data in the current airline dataset are mostly identical; therefore, it is difficult to design classification models in semi-supervised and unsupervised ways. Further analysis regarding the performance of DAP will be discussed in the following sections.

B. Evaluation against SOTA

In this section, the performance of DAP-Supervised (DAP-S) method is evaluated against the baseline FFNN method proposed in [27] using the pre-processed dataset discussed in Section III-A.

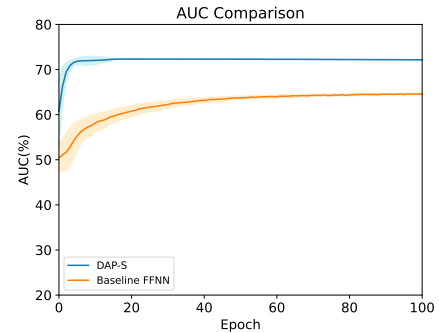


Fig. 6. Mean AUC comparison for DAP-S and the baseline FFNN (SOTA). The shaded error band is the standard deviation of the 10 repeated runs. DAP-S learns quicker and outperforms the SOTA by 10 percentage points.

From Table I, DAP-S outperforms the baseline methods in all metrics while the only difference between the two models

TABLE I
COMPARISON BETWEEN THE BASELINE METHOD FROM [27] AND THE PROPOSED DAP IN A SUPERVISED SETTING.

	<i>AUC</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Baseline	0.621 ± 0.29	0.539 ± 0.02	0.146 ± 0.01	0.654 ± 0.03	0.239 ± 0.01
DAP-S	0.718 ± 0.05	0.698 ± 0.02	0.216 ± 0.01	0.655 ± 0.02	0.325 ± 0.01

is the FE module in DAP-S, which confirms the FE is able to learn a new and enhanced feature embedding during the pre-training stage. Fig. 6 reports the AUC for the test set when the training process is ongoing. With the help of the FE, the AUC for DAP-S reaches the highest at the beginning of the training process, while the baseline method requires a longer time to achieve its best. This proves that by utilising the proposed FE, DAP-S can provide faster inference speed with a shorter training time being used, therefore, improving the training efficiency. Further analysis of the proposed DAP being evaluated against alternative algorithms will be discussed in the following section.

C. Evaluation against Alternative Algorithms

In this section, we benchmark the performance of DAP against AE with loss-thresholding, general anomaly detection algorithms: One-class Support Vector Machine (OCSVM), Local Outlier Factor (LOF), Isolation Forest (I-Forest), and Deep SVDD; and DAGMM. The best algorithm with the highest AUC from the general anomaly detection algorithms is then chosen for further evaluation; in this case, OCSVM with its further results C-OCSVM (using features from AE) and DAP-OCSVM (using features from DAP-AE) are reported, respectively. Note that: (i) for the loss threshold δ , it is defined as the average reconstruction errors calculated by using the training set; for OCSVM, we applied kernels of *linear*, *poly*, *rbf*, and *sigmoid*; (iii) for LOF and I-Forest, different number of neighbours and estimators are tested, respectively; (iv) for Deep SVDD, an AE is pretrained first then the encoder parameters are shared with the main model for further training; (v) for DAGMM, we randomly chose two reconstruction features from Equation 5, 6 and 7; the model is designed as FC(input_dim, 32, tanh)-FC(32, 10, tanh)-FC(10, 1, none)-FC(1, 10, tanh)-FC(10, 32, tanh)-FC(32, input_dim, none), and FC(3, 10, tanh)-Dropout(p=0.5)-FC(10, 2, softmax). The best performance from the configurations of each method is selected and shown in Table II. Regarding the AE structure, *Enc*($\cdot$) is fixed as (input_dim, 64, ReLU)-FC(64, 32, ReLU)-FC(32, 10, ReLU)-FC(10, 1, ReLU), and *Dec*($\cdot$) is the reverse of *Enc*($\cdot$).

Note that one of the key aspects of DAP is the feature compression ability from AE; therefore, an ablation test for the AE module is performed and also shown in Table II, i.e., *FE-LOF*, *FE-I-Forest*, *FE-OCSVM*, *FE-Deep SVDD*, such that the feature embedding from the FE is directly used by the benchmark algorithms. It is clear to see that by using the feature embedding, all the general anomaly detection algorithms have better detection performance in terms of Precision, Recall, and F1-Score compared to the ones using the original

data. However, the overall performance of the proposed DAP framework still outperforms the algorithms that only utilise the feature embeddings, showing the importance of the AE module.

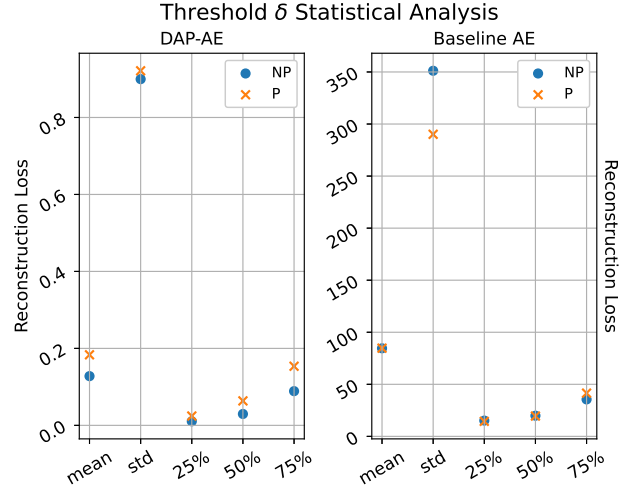


Fig. 7. Statistical analysis of the loss-thresholding method for DAP-AE and baseline AE. The threshold δ is calculated as the mean reconstruction errors of the training set, where *mean* (x-axis) is the average reconstruction errors for *P* and *NP*; it also measures the classes difference. *std* is the standard deviation of the reconstruction errors for the two classes. 25%, 50%, 75% are the percentiles for each of the classes.

From Table II, DAP is able to provide significant improvement compared to the benchmark algorithms; this is due to the FE's ability to learn and generate more distinct features for the downstream AE-based models to perform the prediction. Fig. 7 depicts the statistical analysis of the loss thresholding method for DAP-AE and baseline AE using the reconstruction errors for both class *NP* and *P*. Note that the rule *C* and the threshold δ are correlated to the reconstruction error; DAP provides a better decision boundary as the two classes are more separable in DAP compared to the baseline AE. The reconstruction process is also more stable than general error-based AE methods as a bigger standard deviation of the reconstruction errors is found in the AE-based method. As for general anomaly detection algorithms, the unsupervised-based methods, i.e., LOF, I-Forest, and Deep SVDD did not perform well, possibly due to similar feature properties in both classes.

V. CONCLUSION

In this paper, a multi-stage framework DAP is proposed, which combines a contrastive learning-based feature extractor and several supervised and semi-supervised methods for the task of airline ancillary prediction. The proposed framework

TABLE II
COMPARISON BETWEEN THE PROPOSED DAP AND BENCHMARK ALGORITHMS.

	<i>AUC (%)</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
LOF	50.23 $\pm$ 0.15	0.847 $\pm$ 0.002	0.112 $\pm$ 0.005	0.060 $\pm$ 0.004	0.080 $\pm$ 0.05
I-Forest	49.99 $\pm$ 0.03	0.889 $\pm$ 0.0004	0.045 $\pm$ 0.06	0.0005 $\pm$ 0.007	0.0009 $\pm$ 0.001
OCSVM	53.11 $\pm$ 0.23	0.519 $\pm$ 0.002	0.119 $\pm$ 0.001	0.524 $\pm$ 0.007	0.194 $\pm$ 0.002
Deep SVDD	48.90 $\pm$ 0.005	0.687 $\pm$ 0.0002	0.217 $\pm$ 0.01	0.007 $\pm$ 0.0003	0.014 $\pm$ 0.0008
FE-LOF	49.30 $\pm$ 0.23	0.583 $\pm$ 0.002	0.296 $\pm$ 0.004	0.261 $\pm$ 0.005	0.277 $\pm$ 0.004
FE-I-Forest	50.22 $\pm$ 0.40	0.878 $\pm$ 0.004	0.145 $\pm$ 0.05	0.019 $\pm$ 0.01	0.034 $\pm$ 0.01
FE-OCSVM	55.16 $\pm$ 0.16	0.532 $\pm$ 0.002	0.348 $\pm$ 0.001	0.602 $\pm$ 0.003	0.441 $\pm$ 0.0005
FE-Deep SVDD	47.79 $\pm$ 4.58	0.689 $\pm$ 0.002	0.277 $\pm$ 0.09	0.009 $\pm$ 0.003	0.017 $\pm$ 0.01
AE	50.81 $\pm$ 1.62	0.638 $\pm$ 0.02	0.333 $\pm$ 0.009	0.180 $\pm$ 0.06	0.228 $\pm$ 0.05
DAP-AE	59.69 $\pm$ 2.56	0.644 $\pm$ 0.01	0.369 $\pm$ 0.04	0.234 $\pm$ 0.05	0.285 $\pm$ 0.05
C-OCSVM	50.71 $\pm$ 1.51	0.504 $\pm$ 0.01	0.310 $\pm$ 0.009	0.501 $\pm$ 0.02	0.383 $\pm$ 0.01
DAP-OCSVM	64.96 $\pm$ 10.01	0.566 $\pm$ 0.04	0.384 $\pm$ 0.05	0.714 $\pm$ 0.13	0.500 $\pm$ 0.07
DAGMM	49.68 $\pm$ 0.97	0.615 $\pm$ 0.01	0.304 $\pm$ 0.006	0.198 $\pm$ 0.01	0.240 $\pm$ 0.01
DAP-DAGMM	67.00 $\pm$ 5.33	0.669 $\pm$ 0.02	0.444 $\pm$ 0.04	0.325 $\pm$ 0.04	0.375 $\pm$ 0.04

solves issues faced by the airline industry, where class imbalance and various statistical data properties are too similar for further application of basic anomaly detection algorithms. In our thorough experiments, DAP is able to train the feature extractor module and the ancillary prediction module separately. This combination allows the ancillary prediction unit to achieve great prediction performance with the help of the rich features learnt by the extractor module.

Future work will target to improve both the feature extractor module and the ancillary prediction module; both modules can potentially benefit from an attention mechanism to enhance the performance [36]. The prediction module can possibly be improved by incorporating convolutional neural networks for feature extraction and compression. The model retraining policy for online deployment is also worth exploring. The framework allows for an adjustable policy to dynamically retrain the model if a market variation is detected; such variations are defined as the customers' behaviours changing due to various factors, such as the COVID-19 pandemic. The feature extractor and ancillaries prediction modules can then be updated using the latest available data to incorporate the new market variations. Such behaviours shift can be detected using a method recently presented in [37], where a fast generalised subset scanning and casual forests are proposed to detect the concept drift problem. In our paper, DAP focuses on the prediction of purchasing a single additional ancillary of customers; future work should also consider extending the work to multiple ancillaries provided by the airline companies. In this paper, we only focus on the airline ancillary prediction as most machine learning algorithms did not perform well under this scenario; it is worth to extend the application of the proposed DAP framework to other real-world anomaly detection tasks.

A. Ethical Implications

The proposed DAP framework is a classification algorithm which has the same properties of other published classification algorithms. Customers' personal and confidential information has not been recorded or saved, and no IP tracking or cookie

information has been stored or used for the dataset analysis. The used dataset is GDPR compliant².

ACKNOWLEDGMENT

This work was partially funded by Deepair. The authors would like to thank Deepair for providing support, expertise, the context and the assessment dataset and Deepair's airline partners for their continuing support.

REFERENCES

- [1] Jan K Brueckner, Darin N Lee, Pierre M Picard, and Ethan Singer. Product unbundling in the travel industry: The economics of airline bag fees. *Journal of Economics & Management Strategy*, 24(3):457–484, 2015.
- [2] N.K. Taneja. *Designing Future-Oriented Airline Businesses*. Taylor & Francis, 2016.
- [3] Statista Research Department. Airline industry - ancillary revenue 2011–2021 — Statista, 2021.
- [4] Alex J. Smola and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, 8 2004.
- [5] Tin Kam Ho. Random decision forests. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 1:278–282, 1995.
- [6] Zahid Younas Khan, Zhendong Niu, Sulis Sandiwarno, and Rukundo Prince. Deep learning techniques for rating prediction: a survey of the state-of-the-art. *Artificial Intelligence Review*, 54(1):95–135, 2021.
- [7] Paul Resnick and Hal R Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.
- [8] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, 52(1), 2019.
- [9] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat Seng Chua. Neural collaborative filtering. *26th International World Wide Web Conference, WWW 2017*, pages 173–182, 2017.
- [10] Aäron Van Den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. *Advances in Neural Information Processing Systems*, 2013.
- [11] Jian Wei, Jianhua He, Kai Chen, Yi Zhou, and Zuoyin Tang. Collaborative Filtering and Deep Learning Based Hybrid Recommendation for Cold Start Problem. *Proceedings - 2016 IEEE 14th International Conference on Dependable, Autonomic and Secure Computing, DASC 2016, 2016 IEEE 14th International Conference on Pervasive Intelligence and Computing, PICom 2016, 2016 IEEE 2nd International Conference on Big Data*, pages 874–877, 2016.
- [12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *37th International Conference on Machine Learning, ICML 2020, Part F16814:1575–1585*, 2020.

²For further information, please contact naman@deepair.com

- [13] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A Survey on Contrastive Self-Supervised Learning. *Technologies*, 9(1):2, 2020.
- [14] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 2020-Decem, 2020.
- [15] Okan Kopuklu, Jiapeng Zheng, Hang Xu, and Gerhard Rigoll. Driver anomaly detection: A dataset and contrastive learning approach. *Proceedings - 2021 IEEE Winter Conference on Applications of Computer Vision, WACV 2021*, pages 91–100, 2021.
- [16] Charu C. Aggarwal. *Outlier analysis*, volume 9781461463. New York, 2013.
- [17] Nadipuram R. Prasad, Salvador Almanza-Garcia, and Thomas T. Lu. Anomaly detection. *Computers, Materials and Continua*, 14(1):1–22, 2009.
- [18] Vicente Alarcon-Aquino and Javier A. Barria. Multiresolution FIR neural-network-based learning algorithm applied to network traffic prediction. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 36(2):208–220, 2006.
- [19] Edgar S. García-Treviño and Javier A. Barria. Structural generative descriptions for time series classification. *IEEE Transactions on Cybernetics*, 44(10):1978–1991, 2014.
- [20] Edgar S. Garcia-Trevino, Pu Yang, and Javier A. Barria. Wavelet probabilistic neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–14, 2022.
- [21] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep Learning for Anomaly Detection: A Review. *ACM Computing Surveys*, 54(2), 2021.
- [22] Apapan Pumsirirat and Liu Yan. Credit card fraud detection using deep learning based on auto-encoder and restricted Boltzmann machine. *International Journal of Advanced Computer Science and Applications*, 9(1):18–25, 2018.
- [23] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding Gaussian mixture model for unsupervised anomaly detection. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, pages 1–19, 2018.
- [24] Joao Pereira and Margarida Silveira. Unsupervised Anomaly Detection in Energy Time Series Data Using Variational Recurrent Autoencoders with Attention. *Proceedings - 17th IEEE International Conference on Machine Learning and Applications, ICMLA 2018*, pages 1275–1282, 2019.
- [25] Rong Yao, Chongdang Liu, Linxuan Zhang, and Peng Peng. Unsupervised anomaly detection using variational auto-encoder based feature extraction. *2019 IEEE International Conference on Prognostics and Health Management, ICPHM 2019*, 2019.
- [26] Peter Belobaba, Amedeo Odoni, and Cynthia Barnhart. *The global airline industry*. John Wiley & Sons, 2015.
- [27] Naman Shukla, Arinbjörn Kolbeinsson, Lavanya Marla, and Kartik Yellepeddi. Adaptive Model Selection Framework: An Application to Airline Pricing. 2019.
- [28] Arinbjörn Kolbeinsson, Naman Shukla, Akhil Gupta, Lavanya Marla, and Kartik Yellepeddi. Galactic air improves ancillary revenues with dynamic personalized pricing. *INFORMS Journal on Applied Analytics*, 2022.
- [29] Markus M. Breunig, Hans Peter Kriegel, Raymond T. Ng, and Jörg Sander. LOF: Identifying density-based local outliers. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 29(2):93–104, 2000.
- [30] Fei Tony Liu, Kai Ming Ting, and Zhi Hua Zhou. Isolation forest. *Proceedings - IEEE International Conference on Data Mining, ICDM*, pages 413–422, 2008.
- [31] Lukas Ruff, Robert A. Vandermeulen, Nico Gornitz, Lucas Deecke, Shoaib A. Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 4393–4402, 2018.
- [32] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019.
- [33] Pedregosa Fabian, Vincent Michel, Gaël Varoquaux, Bertrand Thirion, Vincent Dubourg, Alexandre Passos, VAROQUAUX PEDREGOSA, Matthieu Perrot, Olivier Grisel OLIVIERGRISEL, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Jake Vanderplas, David Cournapeau, Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Bertrand Thirion, Olivier Grisel, Vincent Dubourg, Alexandre Passos, and Matthieu Brucher. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [34] Jimmy Lei Ba and Diederik P. Kingma. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–15, 2015.
- [35] Geoffrey Hinton and Laurens van der Maaten. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.
- [36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017-Decem:5999–6009, 2017.
- [37] Abhinav Garg, Naman Shukla, Lavanya Marla, and Sriram Somanchi. Distribution Shift in Airline Customer Behavior during COVID-19. 2021.