Forecasting Unstable Approaches with Boosting Frameworks and LSTM Networks

Darío Martínez, Antonio Fernández
Pablo Hernández and Samuel Cristóbal
Innaxis Research Institute
Madrid, Spain
Email: {dm,af,ph,sc}@innaxis.org

Florian Schwaiger
Institute of Flight System Dynamics
Technische Universität München
Germany
Email: f.schwaiger@tum.de

José María Nuñez
and José Manuel Ruiz
Iberia airlines
Madrid, Spain
Email: {jmnunez,jmruizn}@iberia.es

***Abstract*—This paper presents a machine learning algorithm trained to predict unstable approach events. Predictive modeling for unstable approaches (UA) forecasting needs a precursors analysis to determine the most important indicators (features) of aircraft instability. However, since the definition of aircraft instability is entirely dependent on the airline, these precursors might change according to the applied criteria. Most of the times, these precursors are related to the operation, ATC instructions, nearby weather conditions or even specific procedures for the selected airport or runway. We approached UA predictive analysis scenario from two different perspectives aligned with the same objective. On one hand, we performed the precursor analysis and binary classification using machine learning ensemble methodologies (boosting frameworks). On the other, we analyzed the FDM temporal series with Deep Learning techniques, using neural networks with Long Short Term Memory (LSTM) layers to binary classify if an unstable approach was about to happen and to detect unseen hazards or anomalies present in approach procedures.**

***Keywords*—Safety, unstable approach, deep learning, LSTM**

I. Introduction

Approach and landing procedures are some of the most complex procedures in airline operations. Unstable approaches (UAs) account for most approach and landing accidents, and UAs are the causal factor in over the half of all approaches and landing accidents worldwide [1].

Airlines and safety agencies have published stabilised approach criteria that crew need to follow. According to this criteria [2], landing aircraft should be stabilised by reaching 1000 feet above airport level (AAL) (or at about 3NM from runway threshold) under instrument metereological conditions (IMC) or 500 feet AAL (or at about 1.5 NM from runway threshold) under visual meteorological conditions (VMC). Otherwise a go-around is mandatory. An illustration of the approach criteria is presented in Figure 1.

If the criteria is not met, the flight crew must abort the approach and perform a go-around, adding significant operational cost to the flight, increasing complexity and workload for the air traffic controllers (ATCOs), and also reducing runway efficiency for the arrival airport. Solving this problem will not only benefit airlines by improving flight safety, but will also benefit ATCOs by improving runway efficiency and reducing complexity.

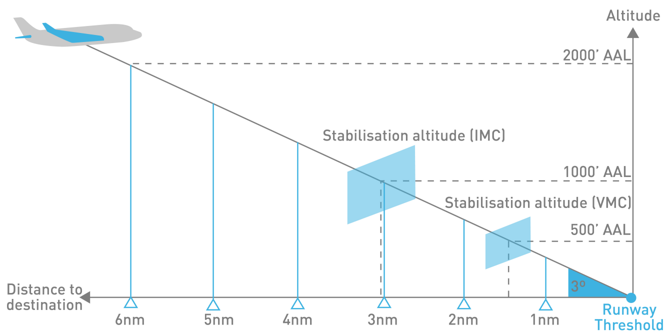

Figure 1: Stabilised approach criteria checkpoints

The broad definition of an Unstable Approach (UA) is an approach that is flown in a manner unconducive for safe landings. With varying definitions of said approach, the end definition is ultimately dependent on airline operational policies. However, they all have a similar set of indicators in common:

- High energy approach: vertical speed and approach speed deviation.
- Exceeding flap/slat limit speed during approach.
- Deviation from intended path angle, glide slope, localiser.
- Excessive attitudes.
- Excessive Tailwind and crosswind.
- Use of speed brakes below 1000 feet AAL.
- Configuration at time points (late gear extension, late flap setting).
- Excessive fan speed during approach.

Using a predictive model to monitor the approach and forecast the destabilisation of the aircraft, alerting the crew seems like feasible approach given the vast amount of flight data monitoring (FDM) datasets stored by airlines. However, currently used indicators in the FDM system are used for tracking the **current state** of the flight, without any predictive capability.

Also, most airlines do not use predictive systems to identify and analyse UAs. Instead, forensic analysis are performed after the flight, using FDM software that labels the data based on a predefined criteria. Manual review and analysis are performed for each flagged flight, and precursors are extracted for each individual case. To date, safety analyst do not use data-driven automatic pattern recognition tools.

The presented paper aims to summarise the progress and results made in the H2020 project Safeclouds.eu to find data-driven prediction models based on contributing factors. We aim to improve the understanding of UA cases with the help of historical data analytics and machine learning or deep learning algorithms, trained with massive amounts of FDM data.

To summarize the lines of research. In this paper, the main research questions (RQ) to be answered will be two:

*RQ1: How **precisely** an UA event can be **predicted** (before occurring) at a certain point of the approach?* 1 2*RQ2: What are the **main precursors**, situations and patterns the contribute to the occurrence of an UA event?*

II. State of the Art

The concept of stabilised approach is described in the flying guide presented by Turner (2011) [3], with specifications and instructions before and after the final approach fix (FAF). It mainly focuses on landing configuration (flaps set and gear down) and aircraft's air speed reduction thresholds. The set of stabilised approach criteria proposed by the Flight Safety Foundation [4] also established the 1000 feet and 500 feet AAL checkpoints that are normally recommended for setting regulations.

Currently, there have not been agreements on the proportion and criteria for UAs. Auditories and studies [5] [6] indicate that around 3% of today's flights are unstable, with only 5% of them initiating the recommended go-around procedure. Wang et al. (2015) [7] conducted a study with statistics of unstable events using surveillance track data, extracting a pattern in operations with 10 knots change in aircraft's ground speed after reaching 1000 feet AAL.

The specific literature related to unstable approach forecasting is not very extensive. Zhenming Wang (2016) [8] proposed the first approach for automatically identifying and "now-casting" unstable approaches, based on a Flight Management Systems (FMS) analysis performed over a small set of training flights data. This approach mainly used conditional probability methodologies and simple supervised learning algorithms and was still able to obtain promising predictive accuracy of more than 80% in a small testing dataset.

Relevant work can be found in related areas such as trajectory prediction, anomaly detection and approach performance analysis. Trajectory prediction is one of the main application fields of machine learning research in aviation. For example, DART (Data-driven Aircraft Trajectory Prediction Research) is a project from SESAR 2020 Exploratory Research with some notable results [9].

However, the main problem with the trajectory forecasting models is that they are training using mainly surveillance data (ADS-B/CPR). This approach may be well-suited for studying en-route events, but it is not adequate for capturing the complexity of the approach and landing procedures. FDM data is required to better explain the aircraft dynamics and changes in the aircraft configuration (e.g. flap/slat dynamics, gear down, etc...).

Regarding anomaly detection, Li et al. (2013) [10] applied a classification method for anomaly detection in FDM data using Multiple Kernel Anomaly Detection (MKAD). Also in [11] (2014), Matthew et al. used surveillance track data to detect en-route anomalies. Due to the unsupervised nature of the algorithms and manual inspection requirements, these anomaly detection techniques are not the most suitable approach for detecting UAs.

There are other methodologies for assessing risk at the approach and landing phase. Some of them are very relevant for understanding the dataset integration, features engineering (i.e. aircraft variables characterization) and specific procedure studies (e.g. go-arounds). Zhang et al. (2010) [12] made an study on the cross-sectional position distributions of arrival flight tracks along glide-path at different airports. They used several types of probability density functions to model position distributions, in which the normal distribution generally provides a good fit.

III. Problem Assessment

As presented previously, it is known that the exact point of trajectory where flights become unstable is near **3NM before the threshold**, giving the pilot between 30 and 90 seconds to react. As such, for a predictive modelling algorithm, the prediction point must be selected between 4NM and 9NM, so that the pilot has enough time to react. In this prediction scenario, the pilot has around 30 seconds to react and opt to perform a go-around.

A very low count of UA events is expected. Therefore to tackle a highly imbalanced predictive problem, we must ensure that our models hit the following requirements:

- Maximise the number of UAs correctly predicted and minimise false negatives. This means avoiding situations where the model predicts an unstable flight that could have landed normally. This has lot of implications in increasing throughput and air traffic operations. This issue highly impacts model confidence when positive predictions turn out to be false alarms.
- Due to the presence of the already known imbalanced problem (only around 5% UAs), the predictive model will probably have to deal with a great amount of false positives. This can cause missing potential unstable approaches that were unlearned during model training. In particular, the model need to deal with the lack of certain situations of UAs in the data or selected features that do not provide enough information on the causes of the instability.

IV. Data

The dataset used is composed by 64.461 approaches performed on 89 European airports. This dataset was created by merging 1 year of FDM, METeorological Aerodrome Report (METAR) and Automatic Dependent Surveillance – Broadcast (ASD-B) data. While FDM as the main data source, ADS-B was used for calculating the traffic throughput at destination

TABLE I: Datasets and features

Group	Features	Data Sources
Operation dynamics	Pitch, roll and heading positions and rates. Angle of attack. Vertical descent rate. Barometric altitude. Glideslope. Localiser.	FDM
Aircraft energy	Air speed. Ground speed. Standard altitude. Energy level. Aircraft mass.	FDM
Adverse weather	Static pressure. Static temperature. Relative humidity. Air density. Wind direction. Wind speed. Wind variation. Prevailing visibility. Cloud layers height. Cloud layers opacity. Phenomena (fog, snow, storms, ...).	METAR FDM (aircraft sensors)
Aircraft configuration	Flaps configuration. Slats configuration.	FDM
Crew coordination	Autopilot status.	FDM
Pilot awareness	Current time. Distance from origin. Distance to destination. Total time flown. Number of holdings	FDM
Surrounding traffic	Airport throughput. VHF keying (tower communication indicator).	ADS-B FDM (communication indicators)
Flight static information	De-Identified callsing. Origin Airport. Destination Airport. Aircraft type. Wake vortex category. Tail number. Year and week. ETA. ATOT.	Flight Plan FDM

airport and METAR to complement information on the weather conditions during the approach.

FDM data presents very high variable dimensionality with more than 150 variables stored as time-series, with a resolution of up to 8 samples per second. Due to this properties FDM data, an initial feature selection based on the case study operational context must be performed. the features can be divided into several groups which each influence unstable approaches based on safety analysts experience. These groups are shown in Table I.

In case of aircraft energy, we extract all values that deal with either speeds or altitude. At this stage, we do not combine them yet, as this should be done at the feature selection stage. There is only one single derived value: energy level. Note that we do not use a geometric altitude here but the barometric, since it is more easily available:

$$e_{total} = \frac{1}{2}V_{kin}^2 + h_{geo} \propto \frac{1}{2}V_{ground}^2 + h_{baro} \tag{1}$$

For the METAR weather data, we not only compute the wind direction and speed, but also its variation. This should hint at changing weather situations or even wind shear phenomena. The METAR report further includes the reported wind direction and speed as gusts and precipitation.

Concerning the crew, we extract which pilot was flying from indicators in the controls. For example, if the left auto pilot, the left sidestick or the right transmitter are active, the captain was most likely pilot flying. The inverse holds for the first officer. We further capture their awareness by calculating the duration of the flight and how many holding patterns performed. To count the number of holdings, we integrate all aircraft turns in either direction and check if the total is divisible by 360 within 5 minutes.

We also suspect that the overall traffic situation at the target airport influences the pressure put on the pilots and, in turn, approach quality. To measure that, we first capture the density in air traffic control (ATC) from the radio communication activity (VHF), a binary time-series registered in FDM. To get a closer look at the situation in the approach chain, positions track data from ADS-B can be analysed at the time shortly before landing. We consider the situation with the leading/trailing aircraft relevant and measure separation distance and speed difference.

V. Methodology

Using purely predictive methodologies based on data science, we tackled the problem from two different perspectives that try to answer the presented research questions. On one hand, we performed a precursor analysis and a binary classification using **Gradient Boosting frameworks**. On the other, we analysed the FDM temporal series with **deep learning** techniques, using neural networks with Long Short-Term Memory (LSTM) layers to forecast if an unstable approach is about to happen and to detect **unseen hazards** and anomalies present in certain approach procedures:

- **Approach A: machine learning GBM classification**: How accurately can we predict a flight's stability, giving enough time to the pilot to stabilise the aircraft or initiate a go-around before a potential UA? Can we use machine learning to predict this event in advance and extract the main precursors to an UA during descent?
- **Approach B: deep learning LSTM classification**: How effectively can we prevent an UA event, extracting the features directly from the FDM time series? Can we use deep learning based on Long Short-Term Memory networks (LSTMs) to predict if a flight will be unstable, learning from the evolution of past observations in time?

A. Predictive modelling using a Gradient Boosting machine

1) Machine learning problem and labelling: From a machine learning perspective, this case study is considered a **binary classification** problem as our target variable indicates if the flight becomes unstable (positive case) or not (negative case) during the approach. However, due to UAs only occurring around 2% of the time, the labeled dataset is known to be highly **imbalanced**.

The imbalanced data problem has been the main consideration for the machine learning algorithm selection. Furthermore, not addressing this insight correctly will yield an inadequate model training, ensuring many false positives. False positives ultimately entail the prediction of multiple unstable flights that

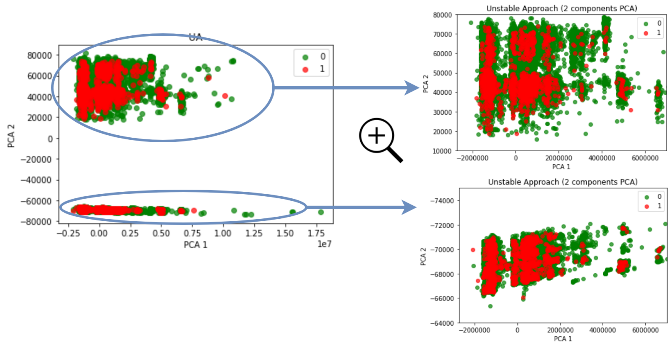

Figure 2: Two-dimensional Principal Component Analysis (PCA) of the dataset. Unstable approaches are the red samples and normal approaches the green samples.

land normally, decreasing dangerously the confidence of our model.

In Figure 2, we present a **Principal Component Analysis (PCA)** of the dataset with 2 components to understand how the target variable is distributed. The PCA visualization gives us two additional insights into the problem. First, we see that classes can't be linearly separable. Therefore, linear discriminant classifiers such as logistic regression, SVM or perception are not sufficient for solving the classification problem. Second, we can observe that 0-1 classes heavily overlap in clusters. Given this distribution and the heavy imbalance of the dataset, identifying and isolating the "1s classes" would be a very complex machine learning challenge. However, given the quality and variety of the datasets used (FDM, METAR, ALLFT+), we can feasibly extract the correct features that identify the classes.

2) Timepoint-based feature engineering: The features for the predictive learning are computed in the same stage with the labelling of UA events. This tight integration is influenced by computational costs, which are mainly driven by reading the flights from storage, pre-processing and flight phase detection, and writing results back to storage. Even using the best machine learning algorithms (boosting frameworks, random forests, adaboost, etc.), it is very difficult to handle the vast dimensionality of FDM data. As a result, features for this have been selected by sampling variables at several points along the approach phase.

However, as most of UA indicators have very short duration limits, usually from 1 to 3 seconds, it can be worthwhile to assess the choice of timestamps. Furthermore, the machine learning model can't perceive the features progression over time, which imposes restrictions on the patterns learned. Taking into account the prediction point 1NM between the prediction point and the moment when UA occurs, we sampled the series beginning from 4 NM to 9 NM, taking a sample every 0.5NM.

Additionally, and for the sake of expanding the precursors analysis, the some interesting time-points on the approach have been labelled and used as features. We engineered features for the following approach time-points that corresponds with the following events: top of descent (TOD), glideslope intercept (G/S ICPT), localiser intercept (LOC ICPT), flaps extended (FLAP 1), flaps full (FLAP FULL), final approach fix (FAF), autopilot disconnect (A/P OFF). These time-points were only considered when they occur between 4NM and 9NM from runway threshold.

In a final remark, each landing attempt/approach is dealt with independently. The destination airport for subsequent attempts might differ, so information about the airport (e.g. runway altitude) cannot be propagated across the whole flight. The same applies to METAR weather reports that can refresh between multiple attempts.

3) Light Gradient Boosting Machine (LightGBM): Gradient Boosting (GB) Frameworks (also known as Gradient Boosting Machines, GBM) [13] are powerful techniques for building predictive models. They select an arbitrary differentiable loss as the objective function and uses an additive model of many weak learners - typically regression trees - to minimise this loss. The parameters of the additional decision trees are tuned by a gradient descent algorithm.

The main advantage of using GBMs over other ML algorithms is that the model is iteratively trained. For each new round, the model uses data samples that were "difficult" to learn in previous iterations. Due to the imbalanced nature of the unstable approach problem, it is considered by the machine learning research community to be one of the most suitable ML algorithms [14].

4) Machine learning results: The normalised confusion matrix (Figure 3) represents how well the model predicted the classes using data not included in the training process. The overall accuracy of classifying the approach as stable or unstable is very good (96,85%). However, notice that the model was able to detect only 85,04% of the unstable approaches contained in the test dataset. This difference can be explained due to the presence of a few false positives samples, and considering the class imbalance.

Features highly increase model performance as far providing meaningful information on the target variable. Figure 4 presents the classifier feature importance, sorted by its impact on determining if a flight is going to be unstable or not.

Analysing the top five features, we can extract some important precusors for the unstable approach events. The variable *weather_altitude_hpa*, which represents the QNH measured in the destination airport, is the most important feature as it combines information from the destination airport (e.g. altitude and position) and weather conditions (pressure variations). The second most relevant feature is the aircraft airspeed at 4NM, which is an obvious precursor for being unstable at 2.5NM. Also, barometric altitude and the airspeed when flaps are fully deployed are relevant and finally the aircraft height descent over time at 4NM. As expected, features around 4NM and 5NM are more relevant than those sampled farther from the point where approach normally becomes unstable.

Furthermore, static features indicating the flight information such as the *tail_number* or *callsign* slightly influence the model. These features often unveil procedures linked to certain airline, crew or aircraft model.

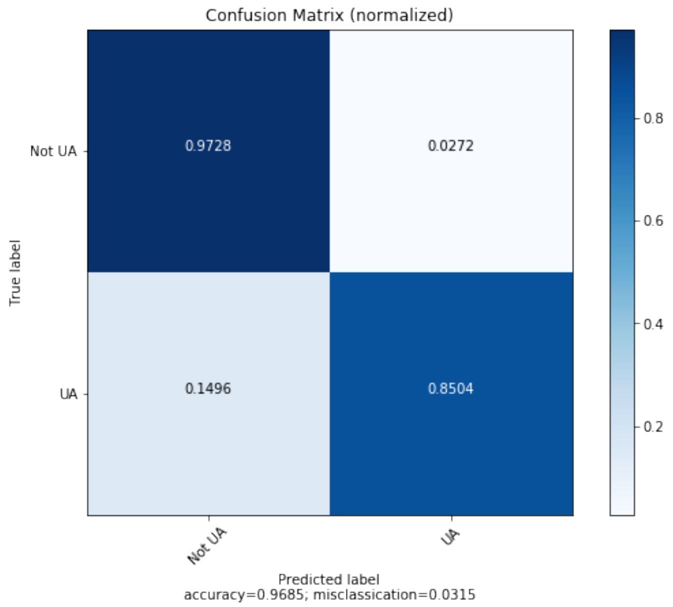

Figure 3: LightGBM classifier confusion matrix

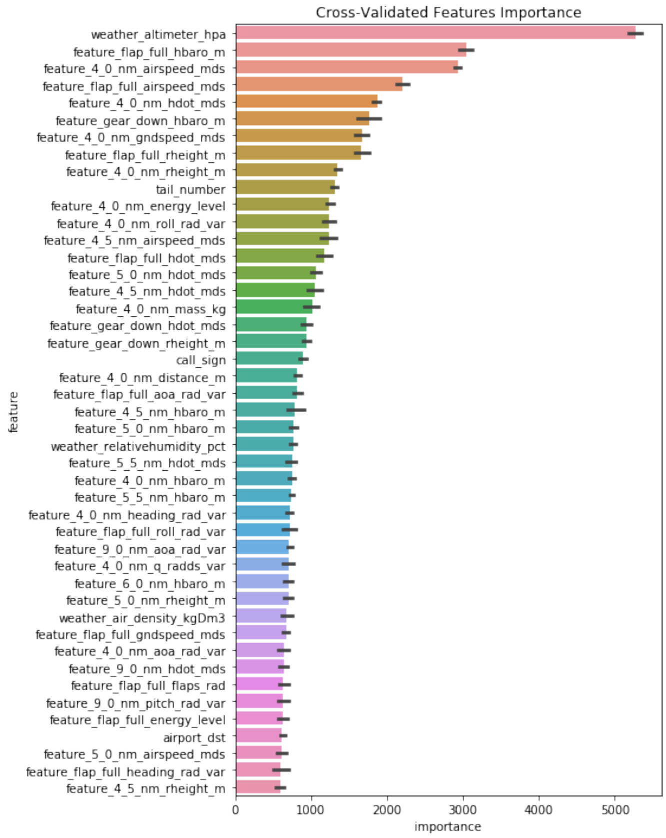

Figure 4: LightGBM classifier feature importance

From the results, we can extract features that were involved somehow in the labeling process, and therefore related with the unstable approach defined criteria as more likely to influence the model, being more relevant at distances closer to the runway threshold. Furthermore, due to the input features being sampled, the model can output the exact point of the sampled time-series that has the most influence for forecasting an UA event. This information can be very useful for pilots in order to make actions (slow down, correct rate of descent, make a go-around, etc...). But, the precursors analysis is also very relevant for airlines safety departments, providing them valuable information for introducing new definitions of UA and internal regulations for the approach operations.

5) Machine learning model limitations: Although the model adequately performs and improves the results presented in the literature, this methodology presents a lot of limitations. First, creating features by sampling the time series every 0,5 NM (30 seconds) means losing large amounts of information. Order of the features extracted from a time series is also not considered, dealing with the features independently which may cause loss of information regarding the evolution of the approach over time.

Also in this methodology, the Prediction point is fixed at 4 NM. Considering unstable approach events being present at 3NM is a wrong simplification that can lead to dangerous malfunctions of the predictive tool. For example, in the event that a UA event happens too soon, this would not give the pilot sufficient reaction time.

Given these reasons, traditional machine learning methodologies are not enough. We need to define a ”dynamic” prediction that also takes into consideration past samples of the time series. In this context, and given the more than sufficient amount of data, it makes sense to shift to a deep learning methodology data, it makes sense to shift to a deep learning methodology.

B. Dynamic prediction using a Long Shot Term memory (LSTM) network

1) Dynamic prediction problem definition and data labelling: We will consider a continuous prediction that ”monitors” periodically, classifying if the aircraft is going to be unstable within a certain future time window. Therefore, we need to define a dynamic prediction point for this case, instead of case A where the prediction was fixed at a 4NM timepoint.

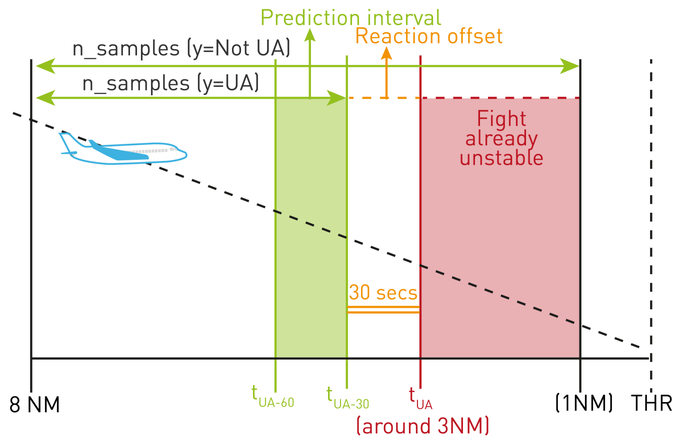

Figure 5: Problem definition for a continuous prediction

TABLE II: Deep learning target variable definition and window interval

Class	Window interval for dynamic sampling
0	From 8NM to the prediction interval end point ($t_{UA} - 30$), having one sample per second
1	From 8NM to 1NM, having one sample per second

Due to the high frequency of the FDM time-series, we will aggregate and synchronise every FDM variable to a 1 second period.

The deep learning model input must be provided with enough observations from the past so that the model can learn how features have evolved before an aircraft becomes unstable. Obviously, the scenario must be constructed so that the prediction is provided before the UA starts or its usage won't be realistic, neither useful.

In case B, contrary to case A, the prediction time point changes for every sample, but it is expected to be around 3NM, excepting outliers and labelling errors. Also, following safety experts and pilots suggestions, the prediction point must be at least 30 seconds before the start of the unstable approach event t_{UA}. Because of this, we must leave a time margin before UA detection point interval $[t_{UA}-30, t_{UA}]$ and shift the labels backwards from this offset $(t_{UA}-30)$, turning the target variable from normal (0) to unstable (1). All the observations after the reaction offset can be removed as irrelevant since the model should not use data from the future in its predictions.

Taking into account the definitions and consideration explained, an illustration on the new problem definition and methodology is presented in Figure 5. The target variable and labelling for this case study is presented in Table II.

2) Feature engineering for the deep learning network: Long Short-Term Memory (LSTM) [15] networks are a type of Recurrent Neural Network (RNN) [16]. They are a very popular choice for making predictions on sequential or time series data [17]. These kinds of models are capable of automatically extracting features from past events and LSTMs are specifically known for their ability to extract both long and short term features using ordered data. LSTM is a bit more demanding than other models referring to data preparation. The input data to an LSTM model is a 3D array, with the following dimensions:

- $n_{samples}$ - refers to the number of observations fed into the LSTM network. In this case, the series has one observation per second.
- $n_{features}$ - the number of columns selected as potential features for the use case.
- $n_{timesteps}$ - (or lookback), describes the time window (past data) needed by the LSTM. To make a prediction at certain time (t), the LSTM will need to process past data up to $(t - n_{timesteps})$.

Given our dataset, we must transform the feature matrix $X[n_{samples}, n_{features}]$ used in for the ML classification into a 3-dimensional array $X[n_{samples}, n_{timesteps}, n_{features}]$. In the 3D array, X, each 2D block at $X[i, :, :]$ denotes the prediction data that corresponds to $y[i]$. To draw an analogy, in regression $y[i]$ corresponds to a 1D vector $X[i, :]$; in LSTM $y[i]$ corresponds to a 2D array $X[i, :, :]$. This 2D block $X[i, :, :]$ should have the at at $input\_X[i, :]$ and the previous rows up to the given lookback. Similarly, this is applied for the entire dataset, for all y's.

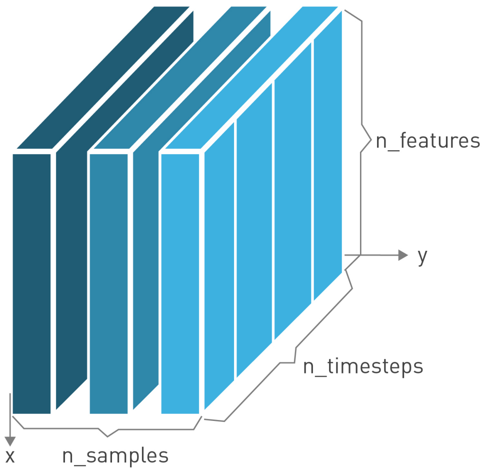

Figure 6: LSTM model dataset input

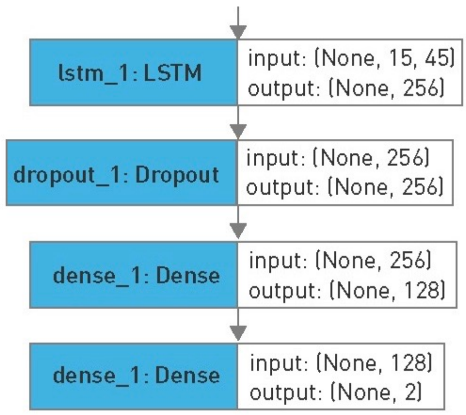

Figure 7: Deep Learning ANN architecture implemented

3) Deep neural network architecture design: Artificial Neural Networks (ANNs) with LSTM layers fit our problem well, enabling direct learning from the temporal series. In addition to the LSTM layers, dense layers are added to help the algorithm learn correlations between features, not only within the samples of the same time-series. To address this learning model, the ANN architecture has been designed from scratch, tuning the network hyper-parameters to properly minimize loss error.

After measuring how much information the LSTM needs from past observations, we finally selected the $lookback = 15s$, which is the number of time-steps needed by the LSTM for each point in the sequence to provide a suitable prediction. It can be observed from the layer architecture diagram (Figure 7) that input size corresponds to the 2D array dimensions $(n\_timesteps, n\_features)$ and the output is composed by two neurons, one for each binary value (1 or 0).

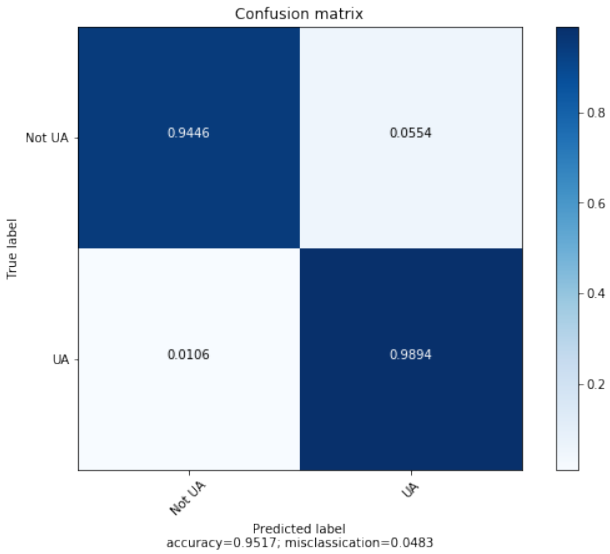

Figure 8: Deep learning classifier confusion matrix

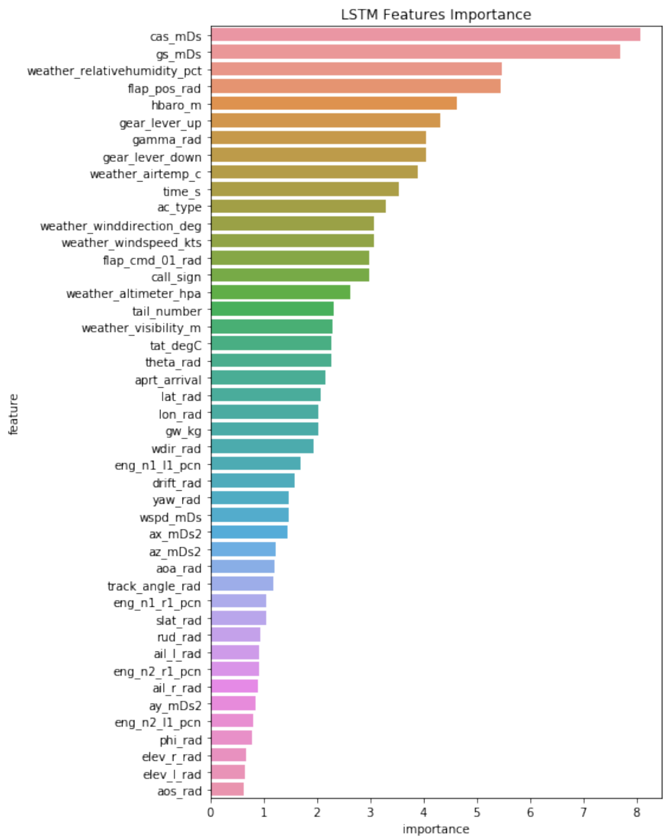

Figure 9: Deep learning classifier feature importance

4) Deep learning results: The normalized confusion matrix in Figure 8 represents how well the model predicted the classes using data that didn't participate in the training process. The model is able to predict most of the points 30 to 60 seconds before an unstable approach is about to occur, having a few amount of false positives that harm UA precision. Although true positive rate have decreased a bit,

This may be related to the close similarity in features of unstable and stabilized flights 60 seconds before a UA triggers, thus causing the model to predict a UA even if the flight finally approached normally. By increasing the amount of unstable flights produced by exceeding other indicators, the model would probably better learn the causes of these types of UA, decreasing the amount of false positives.

The features importance for LSTM neural network is described in Figure 9. The top 5 most important features are very similar to those obtained in case A. The calibrated airspeed is the most important feature, followed by the ground speed.

It must be clarified that while "indicated airspeed" refers to what is read in the pilot static system, calibrated airspeed is the airspeed adjusted for pilot system instrumental errors. Therefore calibrated airspeed is typically within a few knots from indicated airspeed and both can be considered as equivalent for the precursors analysis.

We also appreciate at the top 5 features such as the destination airport relative humidity, the flaps position time-series and the barometric altitude time-series. We can appreciate that in contrast with the other precursors analysis, the features provide less information. Therefore the pilot would be able what kind of variable has the most influence for the UA forecast but he would not know the exact time-point. Because of this, we believe this approach is the most suitable for making a simple alert system. For example, an implementation could be a flag in the cockpit indicating the risk of aircraft being unstable in the next 30 to 60 seconds.

VI. Discussion

If we compare precursors from cases A and B, altitude, flaps and indicated airspeed have the highest impact in both scenarios in being able to predict a potential UA. This makes sense as these features are very related with the labeling criteria, see Table III.

The most relevant precursors are directly related with the unstable approach criteria defined by the airlines. The unstable approach is mostly influenced by **aircraft airspeed**, **flaps positions**, **altitude**, **rate of descent** and **meteorological conditions** of destination airport.

As stated, deep learning LSTM approach is more accurate for predicting UAs with 95% as opposed to the 82% presented by the boosted framework. Also the methodology offers a more sophisticated tool for monitoring safety during the approach and landing phase. This approach could be implemented in a monitoring tool, effectively warning pilots and/or controllers with enough time to react.

However, artificial neural networks models are often considered "black-box" algorithms with limited inter-predictability. As presented, you can extract what FDM time-series are relevant but the exact time-points and events are not known. Therefore, this approach is not desirable for extracting precursors of UA events.

On the other hand, the machine learning approach with boosting frameworks gives more detailed information about

TABLE III: Precursors comparison between the two models

Importance	Precursors in Case A (ML)	Precursors in Case B (DL)
1	$weather\_altimeter\_hpa$: METAR destination airport QNH	cas_mDs: FDM calibrated airspeed
2	$feature\_flap\_full\_hbaro\_m$: Aircraft barometric altitude when flaps are deployed	gs_mDs: FDM ground speed time-series
3	$feature\_4\_0\_nm\_airspeed\_mds$: Indicated aircraft airspeed at 4NM from threshold	$weather\_relativehumidity\_pct$: Relative humidity at destination airport
4	$feature\_flap\_full\_airspeed\_mds$: Aircraft indicated airspeed when flaps are deployed	$flap\_pos\_rad$: Aircraft flaps position time-series
5	$feature\_4\_0\_nm\_hdot$: Aircraft descent rate at 4NM from threshold	$hbaro\_m$: Aircraft barometric altitude time-series

the precursors and allows the input of specific time-points of the time series to study the influence. This makes it a viable tool for forensic analysis and safety hazards detection algorithms.

Nevertheless, the presented models are limited by the training dataset used. They learned from a limited set of routes and approach operations, also using a limited set of aircraft types and meteorological conditions. This means that the predicted results will only be acceptable for flights for which the time-series are similar to the training data.

However, correlation between UAs and the extracted precursors (speed, flats positions, weather conditions, etc...) can be easily generalised for any flight. Therefore, further research can be done in order to asses if model generalisation and extrapolation to different cases is feasible. Furthermore, the current methodology is bound to the input data quality linked to complex data preparation pipelines (e.g. FDM decoding, data merging, etc...) and feature engineering processes.

The machine learning algorithms have been designed in order to only provide a probabilistic forecast of an UA event occurring, additionally giving some causes for that event, using the feature importance of the prediction. This means that the models are not intended to be used as a prescriptive analytics tool, i.e. they will not give recommendations to the flight crew regarding the actions required to avoid the UA event.

This research ensures the possibility of making safer approach procedures by using data-based artificial intelligence system. The use of historical data to train machine learning algorithms is slowly being adopted by the aviation industry. The performance of the predictive models presented in this paper have been beyond expectations. This is probably a step forward for ensuring that aviation safety can be improved with the correct adoption of data science tools.

Acknowledgment

This paper describes the procedures and results of one of the five case studies presented in the H2020 project ”Safeclouds.eu”. The authors would like to thank the ”Safeclouds.eu” consortium for providing the expertise and data necessary to formulate and solve the unstable approach case study.

The authors also thank Iberia airlines safety department for their invaluable inputs and operational knowledge. Bringing a close technical view of unstable approach events and FDM data analysis. Their involvement in the research was crucial for its success.

References

[1] International Civil Aviation Organization. Safety report. 2018.

[2] Tzvetomir Blajev and Capt. William Curtis. Go-around decision-making and execution project. Technical report, Flight Safety Foundation, 2017.

[3] T. Turner. Flying lessons - 111208. 2011.

[4] Flight Safety Foundation. Alar briefing note 7.1 stabilized approach. 2010.

[5] IATA. Unstable approaches: Risk mitigation policies, procedures and best practices. *International Air Transport Association*, 2016.

[6] Aviation Safety Boeing Commercial Airplanes. Statistical summary of commercial jet airplane accidents. *Worldwide operations 1959–2016. Retrieved from www.boeing.com/news/techissues/pdf/statsum.pdf*, 2017.

[7] L. Sherry Wang, Z and J. Shortle. Airspace risk management using surveillance track data: Stabilized approaches. In *8th Integrated Communications Navigation and Surveillance (ICNS) Conference, Dulles, VA.*, 2015.

[8] Zhenming Wang. A methodology for nowcasting unstable approaches. 2016.

[9] SESAR. Driven aircraft trajectory prediction research (dart).

[10] Lishuan Li and R. John Hansman. Anomaly detection in airline routine operations using flight data recorder data. *Report No. ICAT-2013-4, MIT International Center for Air Transportation (ICAT)*, 2013.

[11] Bryan Matthews, Santanu Das, Kanishka Bhaduri, Kamalika Das, Rodney Martin, and Nikunj Oza. Discovering anomalous aviation safety events using scalable data mining algorithms. *Journal of Aerospace Information Systems*, 11:482–482, 07 2014.

[12] Y. Zhang and J. F. Shortle. Comparison of arrival tracks at different airports. In *Proceedings of the 4th International Conference on Research in Air Transportation, Budapest Hungary*, 2010.

[13] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, pages 3146–3154, 2017.

[14] Dehua Wang, Yang Zhang, and Yi Zhao. Lightgbm: an effective mirna classification method in breast cancer patients. In *Proceedings of the 2017 International Conference on Computational Biology and Bioinformatics*, pages 7–11. ACM, 2017.

[15] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation 9(8)*, pages 1735–1780, 1997.

[16] Michael I. Jordan. Attractor dynamics and parallelism in a connectionist sequential machine. *Cogitive Science Conference*, pages 531–546, 1986.

[17] Zachary C. Lipton, John Berkowitz, and Charles Elkan. A critical review of recurrent neural networks for sequence learning. 2015.