I Know Where You Are Going: Predicting Flight Destinations of Corporate and State Aircraft [†]

Marc Jourdan [1], Karolis Martinkus [1], David Roschewitz [1] and Martin Strohmeier [2,3,*]

[1] ETH Zurich, 8092 Zurich, Switzerland; jourdanm@student.ethz.ch (M.J.); mkarolis@student.ethz.ch (K.M.); davidros@student.ethz.ch (D.R.)
[2] OpenSky Network, 3400 Burgdorf, Switzerland
[3] Armasuisse Science + Technology, 3603 Thun, Switzerland
[*] Correspondence: strohmeier@opensky-network.org
[†] Presented at the 9th OpenSky Symposium, Brussels, Belgium, 18–19 November 2021.

Abstract: As data of aircraft movements have become freely accessible on a large scale through means of crowdsourcing, their open source intelligence (OSINT) value has been illustrated in many different domains. Potentially sensitive movements of all stakeholders outside commercial aviation are potentially affected, from corporate jets to military and government aircraft. Until now, this OSINT value was shown only on historical data, where automated analysis on flight destinations has been effective to find information on potential mergers & acquisition deals or diplomatic relationships between governments. In practice, obtaining such information as early as possible is crucial. Hence, in this work, we predict the destinations of state and corporate aircraft on live data, while the targets are still in the air. We use machine learning algorithms to predict the area of landing up to 2 h in advance. We evaluate our approach on more than 500,000 flights during 2018 obtained from the OpenSky Network.

Keywords: crowdsourcing; aircraft tracking; open source intelligence

1. Introduction

Flight data are ubiquitous and freely available on a large scale on the internet. Networks such as Flightradar24 or the research association OpenSky [1] collect practically all air traffic globally and store it for real-time and historical analysis. Recent research has shown that using ADS-B data to track and analyse the movements of high-profile aviation users such as corporate, military and government planes can have a severe impact on the privacy, secrecy and effectiveness of their respective missions [2,3]. The finance industry is known to incorporate such data into their trading models [4] and operational security of military missions has been put into question by sightings on web trackers.

While such automated tracking is now possible on a large-scale and in almost real time, attention has recently shifted to the possibility of using predictive approaches in order to obtain flight movements of such actors even earlier [5]. Work at armasuisse Science + Technology has examined the feasibility of using OpenSky's data to predict the destination of flights while still in the air, potentially yielding actionable open source information earlier than reactive approaches. Using classical machine learning algorithms such as random forest algorithms, aircraft used by different types of stakeholders (military, government, corporate) could be predicted to land with a high accuracy up to two hours before landing.

This paper makes two main contributions:

- Accurate prediction of a plane's destination is possible up to 3 h before landing with a model suited for online prediction;
- Fine-grained estimation of the destination is possible with a manageable drop in accuracy for large number of clusters.

The remainder of this paper is organised as follows: Section 2 provides the necessary background, Section 3 defines the problem of flight tracking, Section 4 our experimental design and Section 5 the results. Finally, Section 6 concludes this work.

2. Related Work

The most popular research question when dealing with flight prediction is to predict the exact trajectory over a given time period. This is of paramount importance for air traffic control in order to monitor, control and guide traffic more easily and efficiently. The common approach is to model the trajectory as an Hidden Markov Model (e.g., [6]). Recently, this research has been extended using Deep Learning, for example by leveraging the performance of LSTM [7,8]. In contrast to our approach, such research mostly focuses on a specific connection between two major cities where the destination is known and may integrate additional data sources such as weather data.

The second popular research avenue with regard to flight prediction is the prediction of the flight delay. This information is important for many stakeholders, from airlines to air traffic management. Existing works aim at predicting the delay distribution with either a statistical approach [9] or a Deep Learning approach [10,11]. A review on the subject is also available [12].

Models have also been developed to understand the behavior of the passengers themselves—the objective being to predict to which airport the passenger will fly given the origin airport [13,14].

Existing works study the privacy impact of large-scale aircraft tracking on governments and public corporation [2,15]. This line of research is the most closely related to ours since we use the same data source and leverage a similar labelling scheme for the categories of the aircraft. To the best of our knowledge, this work is the first to formulate and address the problem of live aircraft destination prediction in this work.

3. The Aircraft Destination Prediction Problem

On a fundamental level, flight trajectories can be represented as a time series of values: position (latitude, longitude, altitude), speed and heading. These time series often contain missing values as they are collected from noisy crowdsourced systems such as the OpenSky Network [16]. Two approaches are possible in this setting, either working directly with the time series, e.g., using Hidden Markov Models or Recurrent Neural Networks, or we manually extract a fixed length feature vector that best represents the flight. In our work, we focus on this second approach and manually create this trajectory embedding.

3.1. Usage Model

As alluded to in Section 1, there are several potential use cases for predicting aircraft destinations in advance. For a military aircraft, the purpose could be to prevent a surprise attack or anticipate the movement of resources. Governmental flight destinations could indicate trade agreements or secret negotiations. The movement of private jets or business flights might foreshadow a potential merger of companies. The information attained through the early prediction of the destination could then be leveraged by a party or counterparty involved.

When adopting the perspective of an attacker, one can distinguish two main categories of attacker: the unsophisticated attacker with none or very limited memory and computational budget and the sophisticated attacker, which can leverage advanced information. In our work, we compare and contrast these two types of attackers, and show that an unsophisticated attacker is sufficient to predict the destination of the flight. An unsophisticated attacker's advantage is that it could be used online (here in the Opensky Network) as it only uses a limited amount of memory and computation. Once the model is trained and stored, the feature vector associated with the flight can be computed easily and evaluated on the fly to have a real-time prediction.

3.2. Research Questions

In light of our usage model, we examine the following research questions:

- How close to its landing can the rough destination of a special interest aircraft (corporate, government and military) be reliably predicted?
- At what granularity can the destination be predicted?
- What features or flight characteristics need to be captured in order to successfully predict the destination?

3.3. Challenges

Predicting the destination of an aircraft carries with it additional challenges but also the potential for further analysis and experimentation.

- No ground truth labels: Solely using transponder data for aircraft that do not file a (public) flight plan means that no ground truth exists. We consider the ground truth to be the actual airport of departure and destination. Often times, the coverage area of the receivers does not reach below a certain altitude which means an aircraft is not tracked all the way to touchdown. We therefore had to devise a labeling scheme to generate training data.
- Granularity of destinations: Worldwide, there are thousands of airports, some of which are frequently used by all class of aircraft, others only in very rare instances. In order to predict the destination, a decision must be made which airports are considered viable destinations, or how to group airports together based on their usage and geographic location. This also allows us to vary the granularity of our approach.
- Destination representation: While the final objective is to predict the destination in terms of an airport (cluster), this could be represented internally as a regression problem with latitude and longitude as the dependent variables. In case the destination is represented as a classification task, depending on the granularity, the number of classes and the class imbalance can be very high. Both approaches allow us to explore slightly different pipelines in the modeling task.

4. Experimental Design

In our work, we focus on interpretability which is difficult to achieve with Deep Learning where the core meaning of each feature is obfuscated, in addition to online scalability, potentially infeasible with big Recurrent Neural Networks.

4.1. Data Sources

We track aircraft in a cooperative way based on their ADS-B transponders using the OpenSky Network. The OpenSky Network [1] is an open-source crowdsourcing platform which started in 2013. It collects raw data from transponders operated by various contributors. Up to now, more than 2 Petabytes of raw data were collected.

The aircraft transponder sends information such as its unique aircraft identifier (the ICAO 24-bit address), GPS position, altitude and ground speed. The raw data are reconciled between different receivers and cleaned up into so-called 'state vectors'. OpenSky also provides a database where state vectors are split into different flights using several heuristics, explained further in [17]. In this paper, we use the flights as split by OpenSky. All these data are stored in an Impala database and made available to researchers and students. The OpenSky Network also provided us with aircraft information such as ICAO24, owner, type, etc. from its aircraft database.

Airport information was gathered from OurAirports in December 2019 (https://ourairports.org).

4.2. Data Pre-Processing

4.2.1. Non-Commercial Flights

From the nearly 500,000 aircraft, multiple layers of filtering were applied to remove commercial (in the sense of airline) aircraft and otherwise to yield only three types of aircraft: governmental, business and military. Heuristics developed for prior research as well as fine-tuning were used to select aircraft based on owner, types of aircraft, callsigns and operators. For example, a Gulfstream private jet was labeled as a business aircraft unless the owner or operator was a known airline, a military or governmental entity or a medical service which might utilize similar aircraft types. Similarly, particular callsigns such as 'SUI' are used for Swiss military aircraft and were therefore labeled as such. Helicopter, traditional piston-powered and single-engine aircraft were also removed. This yielded 20,099 business, 5990 military and 504 governmental aircraft, 26,593 in total.

4.2.2. Geographic Restriction

For each aircraft of interest, every known location was sampled once per minute for the year 2018, given that it appeared in Europe or North America. This was done for data reduction but largely as OpenSky's coverage is best in these areas, in particular in low altitudes.

4.2.3. Data Cleaning

Every state vector has to be associated with a flight or discarded. Therefore, we group state vectors based on the time stamp and split them into different flights when an aircraft has likely been stationary. Outliers in the state vectors are removed (i.e., if altitude is more than 2000 m different from average of two nearby points; if the latitude or longitude jumps by more than 10 degrees; or if the plane is stationary for more than one minute), and gaps of up to 3 min are interpolated. If some information was missing from the state vector (e.g., velocity), we also interpolate it. Then, we compute the first and last *position of interest*, which we define to be midway through departure and approach, respectively. These last and first meaningful positions represent a slice of the flight where most maneuvers are performed. To make training the model easier, we only use information in between these points. If we included any data after the last point of interest, the problem would become trivial because of the proximity of the true airport.

4.2.4. Airport Clustering

We cluster the target airports in order to have control over the granularity of our prediction. Approximately 1000 airports were detected in 2018 for the restricted geographic area. To avoid removing remote airports which are important hubs (e.g., Reykjavik), we used weighted K-means clustering. We tuned it such that airports with few observed flights and no larger airfields nearby are assigned to a distinct leftover cluster. Large airports were also capped in their impact in order to achieve an empirically natural clustering. An example output of this clustering process can be seen in Figure 1.
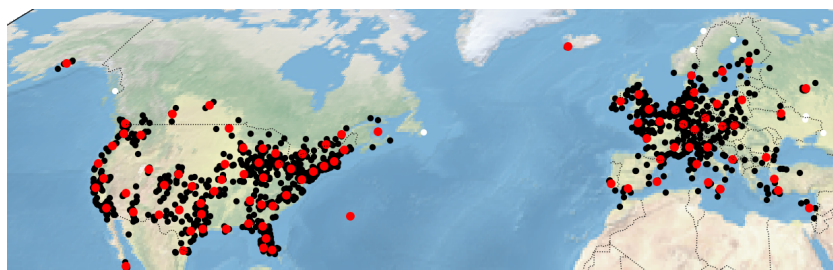

Figure 1. An example of the airport clustering outcome in North America and Europe.

4.2.5. Labelling

As ground-truth destination airports were not provided with the flight data (and are generally not publicly available for typical aircraft of interest), we had to devise a labeling scheme. Here, our priority was on obtaining robust labels, and the procedure is illustrated in Figure 2.
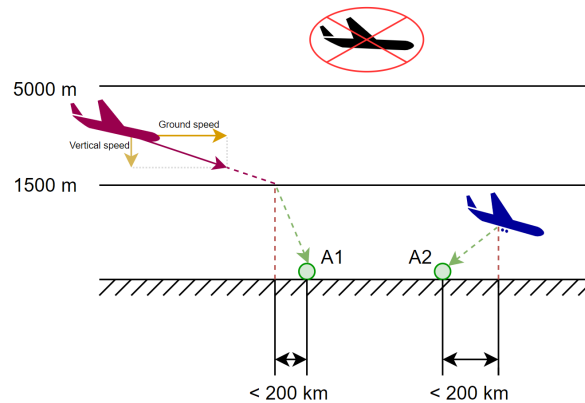

Figure 2. Illustration of the origin and destination labelling process.

For the landing (resp. the departure), we consider the barometric altitude of the last known position (resp. the first). By analyzing the distribution of the barometric altitude over all corresponding positions, we obtain three different zones. First, if the associated barometric altitude of the last position was above 5000 m, we considered that the aircraft was not about to land but simply went to an area without receiver (resp. was not taking off but entered the coverage area during cruise). We discarded all flights without a destination and added 'unknown' for the origin in this case.

Second, if the barometric altitude of the last known position was below 1500 m, we considered that the aircraft's landing was imminent. Hence, for the label, we use the nearest airport, irrespective of heading, as, during the final approach, sharp turns are common.

Finally, in the zone between 1500 m and 5000 m, we use a simple heuristic and consider the first order approximation for the evolution of the aircraft's location. We computed the time taken to reach the 1500-m threshold with the current vertical rate (above $+4$ m/s for the ascent and below -4 m/s for the descent).

4.2.6. Overall Statistics

Note that, throughout this process, any flights or state vectors that could not satisfy all requirements were discarded. For the year of 2018, we started with 23,082 active planes (out of 26,593 interesting planes), which conducted 1,504,931 flights and 119,943,038 state vectors in those flights. After parsing and cleaning, we ended up with 15,287 planes, 729,695 distinct flights and 54,215,760 state vectors.

4.3. Feature Engineering

The main challenge of solving the aircraft destination prediction problem is to encode the available information in a way that allows learning through a classic machine learning approach. For the sake of interpretability and online scalability, we tackle this problem by manually creating a fixed size feature vector per flight.

The simplest approach to encode a trajectory comprises only a single state vector with position (latitude, longitude, barometric altitude), speed (horizontal and vertical), acceleration (evolution rate of the speed information), heading and time (day and minute of the day). We extend this approach to consider multiple state vectors in the trajectory, either randomly selected or equally spaced between the first and the last meaningful position. In practice, we stack the above features for each state vector (except the day).

In addition, we considered other reasonable types of features. First, we added knowledge of the position of the the starting point and by comparison with the current point (delta time and distance). Second, based on first order approximation, we added information on the position of a possible future point (or several) given the current heading and speed at the specific time distance. Finally, we added further features related to the aircraft itself (typecode, ICAO aircraft type, category) and to the departure airport.

Figure 3 shows that the most important features are based directly on transponder information, indicating that an online prediction with restricted sophistication might already provide good results.
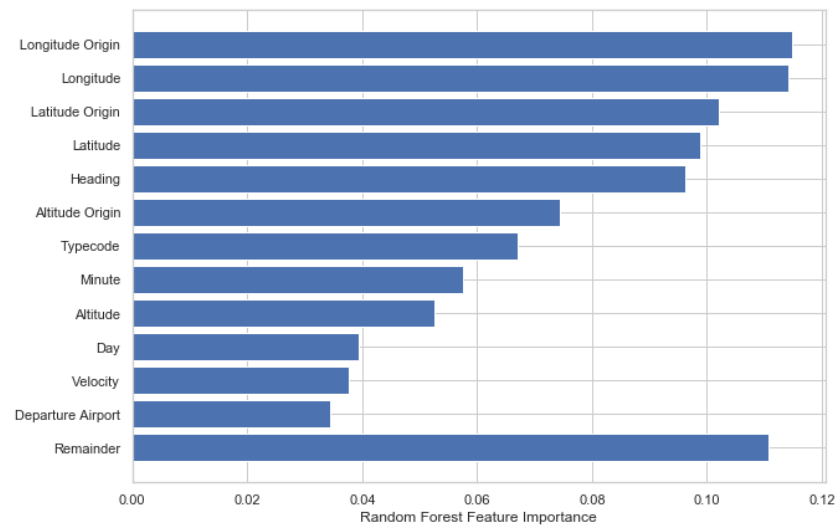

Figure 3. Feature importance.

4.4. Models

As we face a classification problem with a high number of classes, we considered multiple classification models where some feedback on the importance of the features was available and eventually used the Random Forest model for our investigation. To cope with the class imbalance, we trained the tree-based method using the balanced class weighting, which is inversely proportional to the frequency of the class presence. Our main metric is the weighted F1 score, again accounting for class imbalance.

5. Evaluation

We will now evaluate the predictive performance of our model based on three variables: the time before landing, the granularity of the destination, and the type of aircraft.

5.1. Time before Landing

It is common sense to expect that aircraft far away from their destination, here defined in terms of time, are more difficult to predict. As can be seen in Figure 4, within the first 3 h, our model retains an F1 Score above 0.5, and the probability of the true class exceeds that of any incorrect class on average. Results are grouped by 5-min intervals, with 0 indicating that the aircraft has descended below coverage and the landing is imminent.

Here, we also plot the normalized count of data to highlight the non-uniformity of the distribution of data points. We have a disproportionately large number of points corresponding to 30–40 min before landing, and this is mostly an artifact based on our equal sampling procedure and from the fact that some available flights are only constituted of the last hour (flights coming from the area without receivers or short domestic flights). Moreover, we observe that the temporal alignment is not perfect since this time is computed based on the last available point of the flight. This could be when the aircraft is already on the ground but also when the aircraft is still in the air but about to land. Hence, this

temporal proximity from landing is not a perfect indicator, but given the dataset confirms our hypothesis and we can draw valuable conclusions from it.
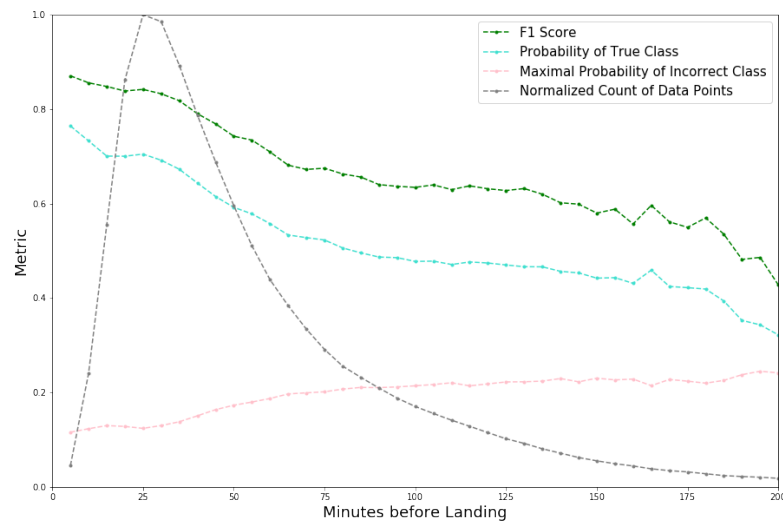

Figure 4. Predictive performance based on time before landing.

5.2. Granularity of Destination

For the vast majority of our experiments, we select 150 clusters based on visual inspection and limited quantitative measures such as the elbow plot. Controlling this value would allow us to influence the granularity at which we can predict the destination. A trade-off between the number of clusters, in essence how fine-grained our prediction aims to be, and performance should exist. With this experiment, we aim to assess this relationship.

Our results in Figure 5 show that the performance deficit is mostly linear. We do not observe a large reduction in predictive performance when we consider 750 clusters instead of 10, hence we can confirm reasonable trade-off between precision and number of clusters.
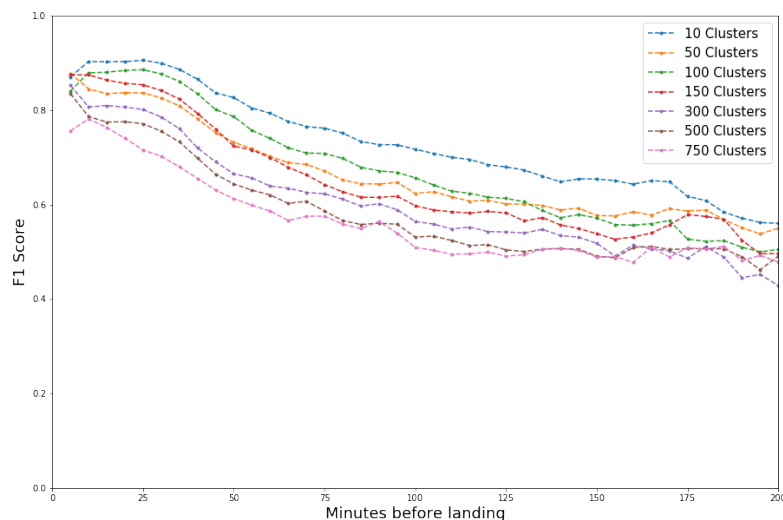

Figure 5. Influence of the granularity on the predictive power.

In a practical scenario, a two-stage approach could be used. First, a coarse model with few clusters is used to obtain with high accuracy the global landing area. Second, a fine-grained model (potentially trained on this particular area) is used to predict the destination at the city or county level. Note that this two-stage approach with models

learned for each specific global area was not implemented but is a suggestion based on our findings.

5.3. Type of Aircraft

Depending on the nature of the flight (e.g., business, military, governmental), we expect the behavior and predictive performance to be somewhat different, with class-specific characteristics based, for example, on the access to different airports.

Our analysis can be seen in Figure 6, and we consider the difference in performance per category for a model trained with data from all those categories. The smoothest curve is obtained for the business jets, resembling the previous generalized results. This is unsurprising as business jets make up by far the largest part of of the flight data.
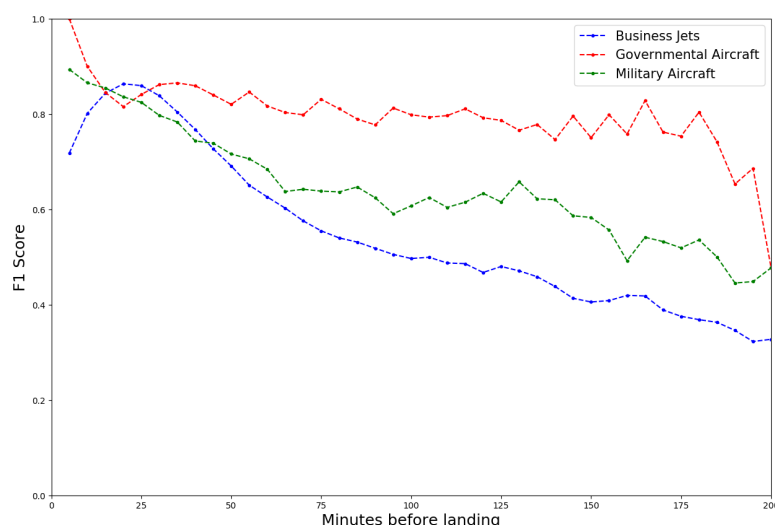

Figure 6. F1 scores predicting the destination cluster for different aircraft types of interest.

The decrease in performance of the military aircraft is similar to the business one, with a slightly slower decay. As for the governmental one, we observe that the performance plateaus before a huge decrease. This could be explained by the fact that governmental flights usually have low variability in terms of destination: they fly to the airports of the capitals of other countries or to cities associated with international institutions (i.e., UN).

6. Conclusions

In this paper, we have motivated and defined the aircraft destination prediction problem. Using a large-scale dataset spanning over 700,000 flights over one year provided by the OpenSky Network, we have shown that it is possible to predict the destination of an aircraft with reasonable accuracy significantly in advance—as much as several hours in some cases. In future work, we seek to improve on this approach through deep learning methods and data sets tailored to specific aircraft types and use cases.

Author Contributions: M.J., K.M., D.R. all contributed equally to this work. Conceptualization, M.S.; methodology, M.J., K.M., D.R.; software, M.J., K.M., D.R.; evaluation, M.J., K.M., D.R.; writing—original draft preparation, M.J., K.M., D.R.; writing—review and editing, M.S.; visualization, M.J., K.M., D.R.; supervision, M.S.; project administration, M.S. All authors have read and agreed to the published version of the manuscript.

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

References

1.　Schäfer, M.; Strohmeier, M.; Lenders, V.; Martinovic, I.; Wilhelm, M. Bringing up OpenSky: A large-scale ADS-B sensor network for research. In Proceedings of the 13th International Symposium on Information Processing in Sensor Networks (IPSN-14), Berlin, Germany, 15–17 April 2014; pp. 83–94.
2.　Strohmeier, M.; Smith, M.; Lenders, V.; Martinovic, I. The real first class? Inferring confidential corporate mergers and government relations from air traffic communication. In Proceedings of the 2018 IEEE European Symposium on Security and Privacy (EuroS&P), London, UK, 24–26 April 2018; pp. 107–121.
3.　Schäfer, M.; Strohmeier, M.; Smith, M.; Fuchs, M.; Lenders, V.; Liechti, M.; Martinovic, I. OpenSky report 2017: Mode S and ADS-B usage of military and other state aircraft. In Proceedings of the 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), St. Petersburg, FL, USA, 17–21 September 2017; pp. 1–10.
4.　Bachmann, J. *Hedge Funds Are Tracking Private Jets to Find the Next, Megadeal*; Bloomberg: New York, NY, USA, 2019.
5.　Harrington, C.S.; Jonas, B.P.; Czerniakowski, F.; Clark, N.J. A Bayesian spatio-temporal aircraft route predictive algorithm with applications to military operations. In *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications III*; International Society for Optics and Photonics: Bellingham, WA, USA, 2021; Volume 11746, p. 117461B.
6.　Ayhan, S.; Samet, H. Aircraft trajectory prediction made easy with predictive analytics. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 21–30.
7.　Liu, Y.; Hansen, M. Predicting aircraft trajectories: A deep generative convolutional recurrent neural networks approach. *arXiv* **2018**, arXiv:1812.11670.
8.　Shi, Z.; Xu, M.; Pan, Q.; Yan, B.; Zhang, H. LSTM-based flight trajectory prediction. In Proceedings of 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018. [CrossRef]
9.　Martinez, V. Flight Delay Prediction. Master's Thesis, ETH Zürich, Department of Computer Science, Zürich, Switzerland, 2012.
10.　Kim, Y.J.; Choi, S.; Briceno, S.; Mavris, D. A deep learning approach to flight delay prediction. In Proceedings of the 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), Sacramento, CA, USA, 25–29 September 2016; pp. 1–6.
11.　Yu, B.; Guo, Z.; Asian, S.; Wang, H.; Chen, G. Flight delay prediction for commercial air transport: A deep learning approach. *Transp. Res. Part Logist. Transp. Rev.* **2019**, *125*, 203–221. [CrossRef]
12.　Sternberg, A.; Soares, J.; Carvalho, D.; Ogasawara, E. A review on flight delay prediction. *arXiv* **2017**, arXiv:1703.06118.
13.　Hess, S.; Polak, J.W. Mixed logit modelling of airport choice in multi-airport regions. *J. Air Transp. Manag.* **2005**, *11*, 59–68. [CrossRef]
14.　Furuichi, M.; Koppelman, F.S. An analysis of air travelers' departure airport and destination choice behavior. *Transp. Res. Part Policy Pract.* **1994**, *28*, 187–195. [CrossRef]
15.　Strohmeier, M.; Smith, M.; Moser, D.; Schäfer, M.; Lenders, V.; Martinovic, I. Utilizing air traffic communications for OSINT on state and government aircraft. In Proceedings of the 2018 10th International Conference on Cyber Conflict (CyCon), Tallinn, Estonia, 29 May–1 June 2018; pp. 299–320.
16.　Schäfer, M.; Strohmeicr, M.; Smith, M.; Fuchs, M.; Lenders, V.; Martinovic, I. OpenSky report 2018: Assessing the integrity of crowdsourced mode S and ADS-B data. In Proceedings of the 2018 IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), London, UK, 23–27 September 2018; pp. 1–9.
17.　Strohmeier, M.; Olive, X.; Lübbe, J.; Schäfer, M.; Lenders, V. Crowdsourced air traffic data from the OpenSky Network 2019–2020. *Earth Syst. Sci. Data* **2021**, *13*, 357–366. [CrossRef]