Analysis of flight systems with augmented Twitter data

Pooja Ganesh
pganesh4@illinois.edu

Srushti Manjunath
srushti5@illinois.edu

1 ABSTRACT

Twitter sentiment and its direct correlation and/or anti-correlation to features such as delay in flights, increase in cost of travel, a recently happened political event etc may be used to obtain indirect correlation/anti-correlation between the above said features and thus make predictions regarding these quantities. As a first step to this idea, in this project, we demonstrate that flight delay predictions of a particular airline, say, Indigo, can be augmented by the Twitter data which is shown to be correlated to delay data. We use Random Forest Regressor to make delay predictions and draw a comparison between Twitter-less model performance and Twitter-augmented model performance. We find that the mean square error metric used to define loss in a machine learning regressor reduces when the model is trained on twitter-augmented data. The reduction even though minor strengthens our initial idea of using Twitter features to make invaluable predictions concerning the aviation industry. We also suggest some ideas for the future work.

2 INTRODUCTION AND MOTIVATION

Excellent data connectivity has led to an explosion of minable data on social media such as Twitter, which in turn reflects on all aspects of human endeavour - from travel, to weather, to geopolitical events, to price fluctuations in stock market, crop failure, to mention a few. On an average Twitter observes approximately 500 million tweets per day and growing - which indicates the amount of information that flows across different parts of the world daily. Relative popularity of Twitter is brought out in Figure 1 as it stands as one of the most widely used social media platforms today.

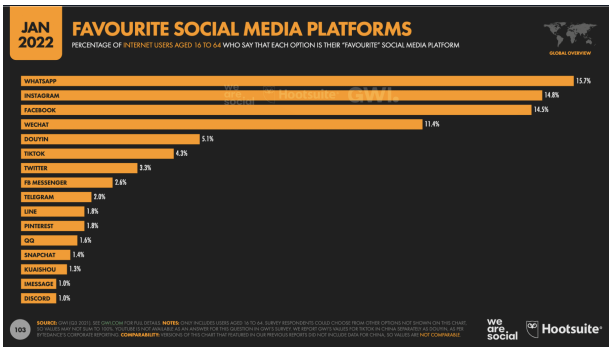

Figure 1: Social Media Platform Popularity 2022

With the advent of data mining and extraction technologies supported by artificial intelligence and machine learning algorithms which are used to make powerful predictions, the question arises - can we augment Twitter data efficiently to improve predictions on parameters such as flight delay and flight fares.

As a first step, we develop a random forest regressor with the twitter-less data. We experiment with four kinds of data distribution styles which had combination of polynomial features, normalized target variable, and regular data. This was done in an effort to improve the model's performance. We find that using polynomial features and a normalized target variable gives the most reduced mean squared error - we use the same procedure for twitter augmented data as well, where our primary idea revolves around using more than the three additional features of polarity, subjectivity, and tweet volume for analysis.

As per our expectations, we find that Twitter can be leveraged to make predictions better. We also discuss the future scope of this project.

3 RELATED WORK

Agarwal's work [1] introduces POS prior polarity for twitter sentiment analysis. Choi et al. [2] have focused on overcoming the effects of the data imbalancing caused during data training. They have used techniques like Decision Trees, AdaBoost, and K-Nearest Neighbors for predicting individual flight delays. A binary classification was performed by the model to predict the scheduled flight delay. Schaefer et al. [3] have made Detailed Policy Assessment Tool (DPAT) that is used to stimulate the minor changes in the flight delay caused by the weather changes.

4 FLIGHT PRICE PREDICTION AND TWITTER DATA NONAVAILABILITY

A more challenging problem to address would be, to be able to predict airfares using information extracted from social media such as Twitter and correlating the same with flight price dynamics which in turn would depend on factors such as Weather forecast, demand, time of the year and geographic trajectory of the flight, eventually leading a flight price prediction tempered by Twitter data. While our preliminary study on flight fare prediction models without Twitter data showed promise, insufficiency of Twitter data related to flight fares during the same timeline as our primary dataset, forced us to look at an alternative but equally challenging problem presented in rest of this project work.

5 PROBLEM STATEMENT

Given weather, airport information from 2019 and an appropriate Tweeter data for the same period, can the delay prediction in arrival of Indigo flight in minutes be augmented to improve performance?

We model this problem as a random forest regression problem where given training data $\{s_i, y_i\}_{i=1}^{n}$, we learn a function f such that the mean squared error loss $\mathcal{L}(f(s_i), y_i)$ is minimized.

We further use additional twitter features to observe model performance improvement.

6 DATASET AND IMPLEMENTATION

6.1 Without twitter augmented data

All models are trained and tested on a Flight Delay Dataset which contains information regarding airport ratings and weather conditions in India for the years of 2018, 2019 and 2020. The entire dataset consists of 14,952 samples, each having a total of 36 feature columns. There are 6 major airlines in India namely Air Asia, Air India, Go Air, Indigo, Spicejet and Vistara. Overall delay is consequence of several other delay factors, for example, Airport Delay (A-Delay), Departure Delay (D-Delay) etc. However, major contribution to the finally delay is known to be primarily from Airport Delay (A-Delay). Hence for this project, we consider the target variable as A-Delay and neglect other delay contributions to the overall delay. Our target variable A-Delay is continuous in nature. A negative delay value indicates a delay by those many minutes. And a positive delay value indicates an early arrival of the flight by those many minutes. In our data, there are 21 weather related features which determine the weather aircraft's environment at the time of its scheduled take-off. The airport ratings comprise come up to 6 features which are a representation of the airport's general reputation performance-wise. They take into consideration aspects such as the airport's on-time and service related performances. The remaining features indicate the aircraft's mechanical performance. Some of the features were dropped in the later stages of the project with the help of feature selection techniques and only keep features relevant to our prediction.

In this project's scope, a smaller dataset of 2298 samples out of 14952 is used. The year of analysis is narrowed down to 2019 from the month of January to December. This choice of time line (i.e, months instead of days or hours) is motivated by relevant Twitter volume. We analyze solely one airline "Indigo" and drop the remaining airlines, though, the method used here is applicable to all the 6 major airlines. The training-validation-testing split is described in table table 1. The delay distribution is shown in the figure .

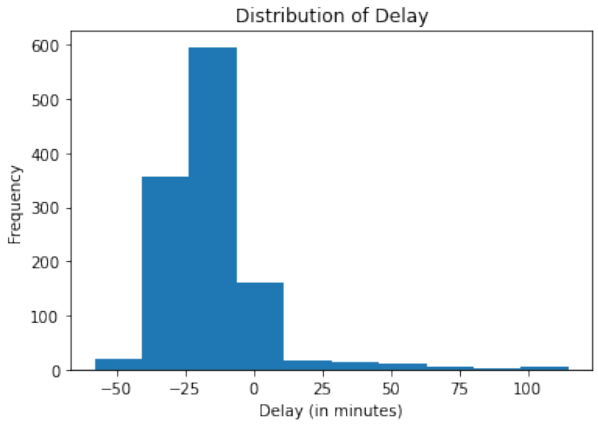

Figure 2: Delay distribution of Indigo Airline in 2019

Split	Features shape	Delay Variable shape
Train	(830, 36)	(830,)
Test	(356, 36)	(356,)

Table 1: Details of the Flight Delay dataset

6.2 With twitter augmented data

A total of 1000 tweets were extracted using certain keywords such as "delay" and "Indigo". Augmentation of three interesting features extracted from Twitter are merged into the existing dataset. The features are "polarity", "subjectivity", and "tweet volume". Their distributions could be understood better with the help of figures 3 and 4.

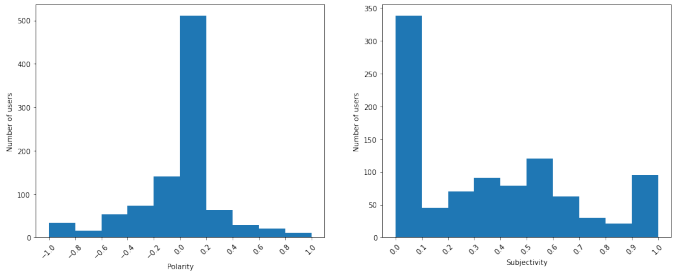

Figure 3: Polarity (left) and subjectivity(right) distribution

Polarity is a measure of happiness observed in user sentiments on Twitter based on their tweets. It ranges from -1 to 1 where -1 implies that the tweets have a very negative connotation to them and 1 implies that the tweets are positive in nature. Similarly, subjectivity ranges from 0 to 1 where 0 implies high objective nature in the tweet and 1 implies high subjective nature of the tweet. The process involved in obtaining the quantities of interests mentioned here will be elaborated in the upcoming sections.

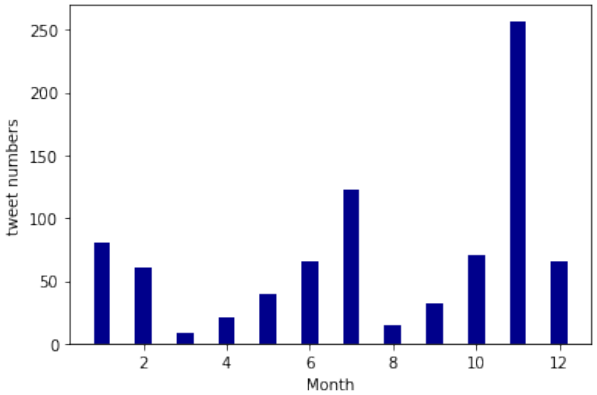

Figure 4: Tweets volume with polarity < 0

Figure 4 shows that there is a hike in the number of negative tweets related to delay in Indigo flight around months of October and November - which does make sense since those are the festival

months in India which are prone to rush hours where flight often get scheduled or delayed to due to the overwhelming number bookings. This could be one way of finding meaning and extent of correctness in our twitter data. It also helps us corroborate the fact that the extracted data is accurate enough for the machine learning model to make sense out it while learning.

7 METHODS AND IMPLEMENTATION

Data cleaning helps make the data cleaner and comprehensible for the model to understand better, thereby improving performance. It involves dealing with missing data, feature engineering, outlier detection and feature selection. A tree based ensemble model is used which uses random forest regressor and support vector machine as its base models to produce one optimal predictive model. The pipeline can be understood more clearly with Figure 5.

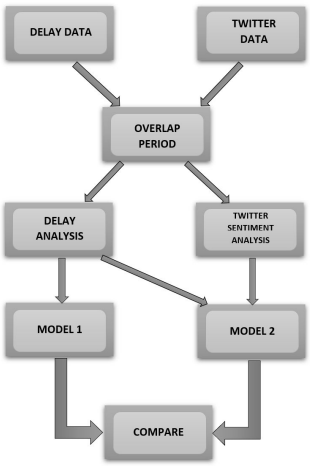

Figure 5: Pipeline depicting all steps in machine learning model development

7.1 Outlier Detection and Target Normalization

7.1.1 Outliers. For outliers, one could say the following: Our focus is primarily the delay data. Figures 1 and 3 suggest that the overall frequency for delays with values is less than or equal to 0 is relatively small, with overwhelming frequency of delays around 25 minutes. Hence we believe that it may not affect the predictions if the entire data is considered as is.

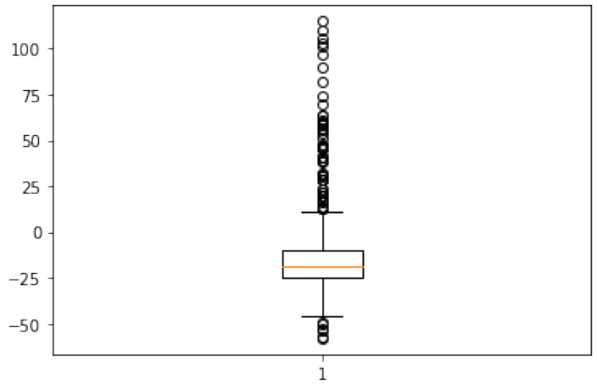

Figure 6: Boxplot for Delays in Indigo in the year 2019

7.1.2 Target Normalization. This is done so that the model generalizes well in real life sometimes. The work done in this projects draws a comparison between model performances which use normalised and unnormalized target variable.

7.2 Feature Engineering

Feature engineering is the process of selecting, manipulating, and transforming raw data into features that can be used in supervised learning.

7.2.1 Feature Importance or Selection. Conducting a feature importance is necessary since it gives an idea about the features which contribute to the model's prediction best. Tree based machine learning algorithms such as Random Forest and XGBoost are equipped with a feature importance attribute that outputs an array containing a value between 0 and 100 for each feature representing how useful the model found each feature in trying to predict the target.

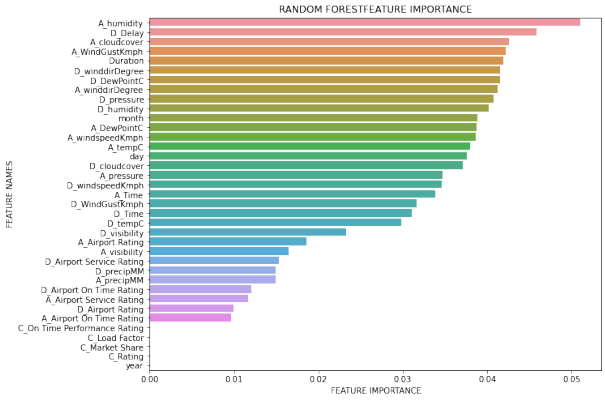

Figure 7: Random Forest Feature Importance

A more clear representation of the top chosen features can be found in Figure 8.

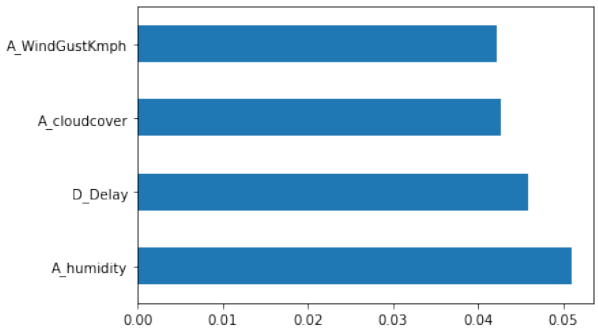

Figure 8: Random Forest Feature Importance

7.2.2 Most important features. It is evident from Figure 8 that the top 4 features which contributed to the accuracy of the model are wind gust, cloud cover, departure delay (D-Delay) and humidity.

7.2.3 Data Leakage. D-Delay emerges amongst one of the top most important features. However, we decide to drop it since it attributes to data leakage. Data leakage is a concept in machine learning used to describe the situation in which the data used to teach a machine-learning algorithm contains unexpected extra information about the target variable being predicted. Since the knowledge about delay in departure would result in model learning a direct relationship between departure delay and arrival delay, this feature does not prove to be very useful. To elucidate further, leakage occurs when information about the target label or number is introduced during learning that would not be lawfully accessible during actual use. We drop departure delay (D-Delay)

7.2.4 Polynomial Features. Often, input features for a predictive modeling task interact in unexpected and often nonlinear ways. Addition of polynomial terms (features) to the model can be an effective way of allowing the model to identify these nonlinear patterns which might lead to improved model performance. Transforms like raising input variables to a power can help to better explore the important relationships between input variables and the target variable. The "degree" of the polynomial is used to control the number of features added, e.g. a degree of 3 will add two new variables for each input variable. We use degree as 2 for the scope of this project.

For example, with two input variables with values x and y and a degree of 2, the features created after a polynomial transform would be:

$$1(bias), x^1, y^1, x^2, y^2, (x).(y)$$

8 MODEL APPLICATION

We draw comparisons between the models trained with the weather data and the model trained with the twitter features augmented with the weather data. Figure 9 below shows the overall idea of what we are attempting in this project.

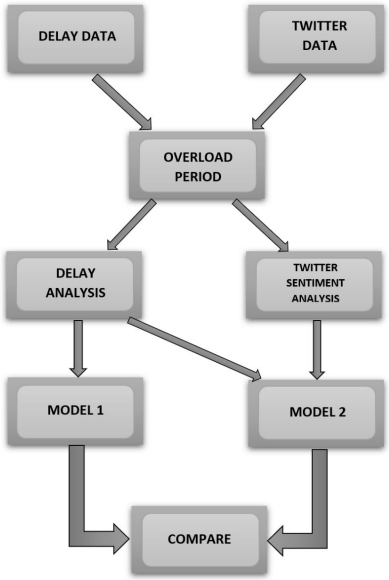

Figure 9: Comparison between augmented and non augmented model - workflow

- Delay data related to weather conditions and twitter share a common time period (date wise), we extract those rows and use that for analysis.
- We conduct sentiment analysis and produce three additional features which will be used in model 2, which we call as augmentation.
- Delay analysis is performed in both models.
- Model 1 predicts delays based on weather information solely.
- Model 2 predicts delays based on weather information as well as augmented twitter features.
- Loss values are compared to infer which is the better performing model, and consequently to understand whether twitter features would be useful addition for prediction.

8.1 Without Twitter-Augmented Data

This section describes the construction and working of model 1 from figure 9.

As mentioned earlier, ensemble model paired with polynomial features gives the best model performance. We optimise our model's performance using power transforms, which are a family of parametric, monotonic transformations that are applied to make data more Gaussian-like. This is useful for modeling issues related to heteroscedasticity (non-constant variance), or other situations where normality is desired. Currently, Power Transformer supports the Box-Cox transform and the Yeo-Johnson transform. Our analysis uses Yeo-Johnson transform, which works with positive and negative values both in contrast to Box-cox which strictly works with positive values. The table shows the mean-squared-error values for the following four cases being fed into the model :

- No change in features, no change in target
- Polynomial features of degree 2, no change in target

- No change in features, target normalized
- Polynomial features of degree 2, target normalized

It is interesting to note that normalizing the target variable might be a good idea looking at the table above. It is clear that we keep the target variable normalized and use polynomial features as our final set of data parameters. Further, we use SelectKBest function to top 60 best features from the polynomial features which are huge in quantity. A support vector machine is used to produce predictions which are again fed into a random forest regressor. This would be our final result.

8.2 With Twitter Augmented Data

This section describes the construction and working of model 2 from figure 9. After the twitter feature merging or augmentation part, the rest of the idea is same as section 6.1.

8.2.1 Extracting Tweets using Twitter API. Extraction of data from Twitter API was done by submission of an academic research application through our developer account where it is required for us to give our project/research details. Upon submission, we gain access to full archive search which we can use to obtain various types of data such as tweets, users, trends, dates, amongst many others. Tweepy library is used for extraction after which we are expected to use our app's consumer key and consumer secret, as well as our access token and access token secret. To elucidate further, we searched for tweets that contain certain keywords for example "delay" and "Indigo" with the help of the search-tweets function. Additionally, we used get-user function to obtain information about a specific user. With academic track access, we made use of full-archive-search function to access tweets backdated to as far as 16 years ago by specifying search-query i.e keywords, a range of date, user fields and other parameters as required. However we have used Tweet data only from the overlapping time period in 2019 for further analysis.

8.2.2 Sentiment Analysis on Twitter features. Sentiment analysis or opinion mining, is a natural language processing (NLP) technique used to determine whether data is of a positive, negative or neutral tone. TextBlob is a python package which is used for this purpose. It's sentiment property returns a named tuple of the form Sentiment(polarity, subjectivity). The polarity score is a float within the range [-1.0, 1.0]. The subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective. In the process of cleaning the newly made twitter dataset, we deal with duplicates and worthless entities in a tweet such as embedded hyperlinks. We used "Levenshtein Distance" metric to find similarities between tweets and remove redundant tweets to account for a clean dataset. It is interesting to understand the functioning of this metric, which was invented in 1965 by the Russian Mathematician Vladimir Levenshtein (1935-2017). The distance value describes the minimal number of deletions, insertions, or substitutions that are required to transform one string (the source) into another (the target). Unlike the Hamming distance, the Levenshtein distance works on strings with an unequal length. The greater the Levenshtein distance, the greater are the difference between the strings. For example, from "test" to "test" the Levenshtein distance is 0 because both the source and target strings are identical. No transformations are needed. In contrast, from "test" to "team" the Levenshtein distance is 2 - two substitutions have to be done to turn "test" in to "team".

After forming a new dataset using these features, we repeat the steps mentioned in section 6.1 and construct a mean square error loss function for our model.

9 RESULTS

The twitter augmented model observes a slight reduction in the mean-square error loss value consistently. As observed in the visualizations below.

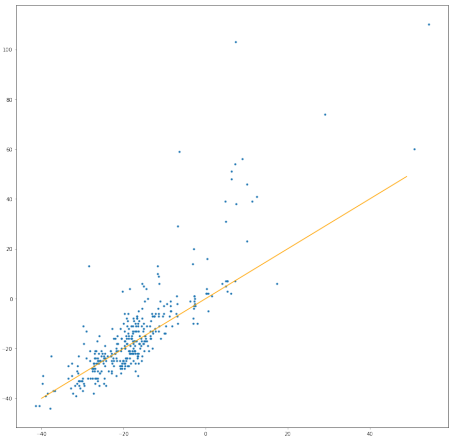

Figure 10: Model trained on data lacking Twitter features

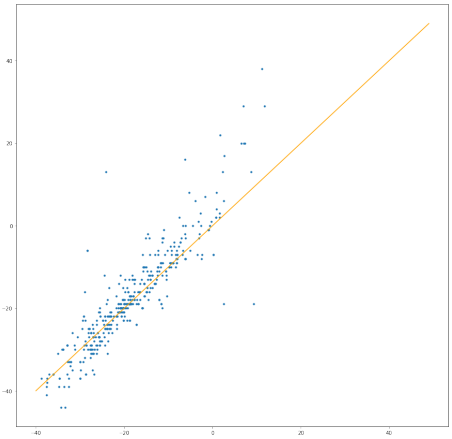

Figure 11: Model trained on data augmented with Twitter features

On first glance, one might look at the plots and observe no tangible changes in the straight line fitting along the scattered points. That is when looking at table 2 would be helpful since it shows a reduction in the mean-squared error values in both the cases of train and test - with twitter augmented data.

Case	Train MSE	Test MSE
No Twitter	0.175	0.290
Twitter	0.089	0.168

Table 2: Features

10 CONCLUSION

The idea of using real-world data from a seamlessly never-ending source of information such as Twitter has endless potential. We tried to exploit what forms the tip of the iceberg for this idea to provide proof of concept. We demonstrate that flight delay predictions of a particular airline, Indigo, can be successfully augmented by the Twitter data to produce enhanced model results as compared to the predictions made by non augmented flight data.

11 CHALLENGES

Some of the challenges for this project included not being able to obtain overlap data between the dataset we had and twitter extracted data - which made it difficult for us to conduct any correlation based analysis on flight price. Finding optimal keywords for twitter was also quite tricky since the trial and error method exhausted our limited number of tweet related operations that came with the access key.

12 FUTURE SCOPE

It does not come as a shock that the enhancements in our model's performance i.e the mse loss are not significantly huge in this project since we only augment our dataset with three new features. The marginal improvement, however, paves way for further analysis of delay that can be conducted as part of future scope for this project. In this project, a clear correlation is established between delay and the correspond twitter sentiment, which has not been done before. This establishes a basis for very interesting future work - for example, if the delay and prices are correlated, which can be found in existing work done by other authors, then we now have an interesting situation where we can make correlations between flight price and twitter sentiment via delay. To enunciate further, delay and price will be anti correlated whereas delay and twitter sentiment will be directly correlated. Additionally, one can also correlate delay to geopolitical events. This project is the first litmus test to show connection between twitter data and certain important aspects of the aviation industry such as delay.

13 ACKNOWLEDGEMENTS

In the course of this project, we referred to several helpful websites listed below :

- http://www.cs.columbia.edu/ julia/papers/Agarwaletal11.pdf
- https://towardsdatascience.com/data-leakage-in-machine-learning-how-it-can-be-detected-and-minimize-the-risk-8ef4e3a97562
- https://machinelearningmastery.com/polynomial-features-transforms-for-machine-learning/
- https://stackabuse.com/levenshtein-distance-and-text-similarity-in-python/
- https://developer.twitter.com/en/docs/twitter-api/premium/search-api/quick-start/premium-full-archive
- https://medium.com/ai-techsystems/gaussian-distribution-why-is-it-important-in-data-science-and-machine-learning-9adbe0e5f8ac

REFERENCES

[1] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment Analysis of Twitter Data. In *Proceedings of the Workshop on Language in Social Media (LSM 2011)*. Association for Computational Linguistics, Portland, Oregon, 30–38. https://aclanthology.org/W11-0705

[2] Sun Choi, Young Jin Kim, Simon Briceno, and Dimitri N. Mavris. 2016. Prediction of weather-induced airline delays based on machine learning algorithms. *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)* (2016), 1–6.

[3] Zhe Zheng, Wenbin Wei, and Minghua Hu. 2021. A Comparative Analysis of Delay Propagation on Departure and Arrival Flights for a Chinese Case Study. *Aerospace* 8, 8 (2021). https://doi.org/10.3390/aerospace8080212