

Weather-Based Link Prediction for LEO-Satellite Networks using the WetLinks Dataset

Eric Lanfer, Dominic Laniewski, Daniel Otten, Nils Aschenbruck
Osnabrück University - Institute of Computer Science, Osnabrück, Germany
{lanfer, laniewski, otten, aschenbruck}@uos.de

Abstract—The performance of Low Earth Orbit (LEO) satellite networks such as Starlink is volatile and susceptible to weather conditions. In this demo, we explore the potential to predict Starlinks link parameters – precisely the upload and download throughputs and the RTT – based on historical weather data and weather-forecast data. An accurate prediction model would enable the development of weather-aware applications. We train different machine learning (ML) models, compare their performance, and demonstrate that the prediction accuracy for all network parameters is sufficient to predict general trends. Especially, the download throughput can be predicted fairly well.

Index Terms—Link Prediction, Starlink, Satellite Communication, Machine Learning, Weather-based Link Prediction, LEO

I. INTRODUCTION

SpaceX's Starlink is the largest operational Low Earth Orbit (LEO) satellite network today. It promises global internet coverage, spanning more than 5,000 satellites [5]. Since its release to the public in 2020, many studies assessed Starlink performance via real-world measurements in different scenarios. Most works focused on the measurement of network parameters, such as throughput, packet loss, and latency. For example, Michel et al. [6] analyzed TCP and QUIC throughput, as well as latency and packet loss. Ma et al. [4] measure latency, packet loss, and TCP and UDP throughput in urban and rural scenarios. Izhikevich et al. [1] conduct world-wide latency measurements.

In contrast, only a limited number of studies analyzed the impact of weather on Starlink performance [2]–[4]. A first, limited look at the impact of rain has been taken in [2] and [4]. Kassem et al. [2] found that the loading-time of web pages increases with rain. Ma et al. [4] found that the upload throughput is largely unaffected by rain, but the download throughput can decrease by up to almost 50%. Laniewski et al. [3] conduct a first in-depth study of weather impact on Starlink performance. They found that not only rain can decrease the UDP download throughput by up to 40%, but also that clouds likely have a significant negative impact. Other weather conditions, such as temperature or wind-speed, only have a minor impact, if at all. They make the *WetLinks* dataset, containing approximately 140,000 network

performance measurements annotated by high-quality weather data, publicly available.

In this demo, we take the *WetLinks* dataset to explore the potential to predict Starlinks link parameters based on weather data. This includes data from the past, as well as weather forecast. An accurate prediction model would enable the development of weather-aware applications, such as new adaptation algorithms for streaming scenarios that decide based on the weather forecast and the predicted throughput when and in which quality data is downloaded. Our main contributions are as follows:

- 1) We demonstrate the capabilities of a random forest (RF) regressor, a support-vector regressor (SVR), and a multi-layer perceptron (MLP) predictor to predict the Starlink RTT, download and upload throughput based on historical and forecast weather data.
- 2) We show that the RF regressor is the best-performing model. It achieves an R^2 of 0.4 for the download throughput, 0.31 for the upload throughput, and 0.32 for the RTT. While this indicates that there are additional factors other than weather that influence especially the upload throughput and the RTT, we demonstrate that this accuracy is sufficient to predict general trends.

II. APPROACH

In this section, we describe our approach to the prediction problem. This includes the dataset, the data preprocessing chain, the prediction models, and a performance analysis.

A. *WetLinks* Dataset

We train our models on the *WetLinks* dataset [3]. It consists of approximately 140,000 Starlink measurements of relevant network parameters, including the download and upload throughput, the RTT, and packet loss. It was collected at approximately three-minute intervals over the span of six months (Oct. '23–Mar. '24) from two European vantage points: Enschede (NL) and Osnabrück (GER). Furthermore, it is augmented with high-quality weather data collected with professional weather stations placed directly next to the Starlink dishes.

B. Data preprocessing

The *WetLinks* data requires different preprocessing steps for training and validation: (1) *nan* values, *inf* values, and samples that contain erroneous weather measurements are removed. (2)

Eric Lanfer and Dominic Laniewski are co-first authors

ISBN 978-3-903176-63-8 © 2024 IFIP

TABLE I: Classifier Comparison

	Download				Upload				RTT			
	RF	MLP	SVR	Dummy	RF	MLP	SVR	Dummy	RF	MLP	SVR	Dummy
MSE	2448.769	3549.089	3507.931	4101.479	28.604	41.288	41.763	42.547	39.420	57.117	56.417	51.603
MAE	36.068	47.468	46.655	51.415	3.773	4.862	4.720	4.790	4.262	5.550	5.384	5.447
R2	0.404	0.137	0.147	0.000	0.311	0.006	0.000	-0.020	0.322	0.017	0.029	-0.005

It was shown that the measurement in Enschede consistently outperform the ones from Osnabrück by approximately constant factors [3]. In order to use the complete dataset, the samples from Enschede are corrected by the differences of the download, upload, and RTT medians of both measurement sites. (3) Daytime-dependent effects of the throughputs were found in [3]. These effects are removed following the approach proposed in [3]. (4) Outliers are removed based on the 1.5 interquartile range (IQR). (5) It was shown that rain has the most impact on the Starlink performance [3]. However, the dataset contains significantly more samples without rain. This imbalance is removed by using all rainy samples, and randomly sampling the measurements without rain.

The resulting balanced dataset consists of 19,390 samples, having 50% rainy and 50% non-rainy samples. In a final step, we create weather forecast and history features for each weather value for 10, 5, 2, 1 minute windows before and after the current sample and append these features to the data.

C. Prediction Models

This section outlines the prediction models utilized in this study, along with their respective training processes. Additionally, we present the parameters that produced the optimal results for each model. We compare the performance of the following four models:

- Dummy predictor (Dummy)
- Random Forest (RF)
- Multi Layer Perceptron (MLP)
- Support Vector Regressor (SVR)

The *Dummy Predictor* always predicts the median value of the input data. This model serves as a straightforward and naive baseline performance comparison. If a model does not significantly outperform the dummy predictor, it suggests that the model may not be learning useful patterns in the data or that its performance may not be significant.

The *scikit-learn*¹ python package in version 1.3.2 is used to train the RF, MLP, and SVR models. For each model, three separate predictors are trained: one to predict the download throughput, a second to predict the upload throughput, and a third one to predict the RTT. All predictors get the same input feature vector X , containing the following weather features: *humidity*, *wind speed*, *wind direction*, *rain*, *solar radiation*, *atmospheric pressure*, *temperature*. Each of these features is available in past and future forecast directions from 10-1 minute windows, as previously described. This results in total to 63 input features. The hyperparameter tuning of the models

is done by using random search, specifically by using the *RandomizedSearchCV* class with more than 500 iterations per model. The resulting best hyperparameters for each model are as follows:

- RF: `max_depth=105`, `max_features='sqrt'`, `min_samples_leaf=1`, `min_samples_split=4`, `n_estimators=101`
- MLP: `hidden_layer_sizes=(77, 90)`, `activation='relu'`, `solver='adam'`, `alpha= 0.01`, `learning_rate='constant'`
- SVR: `kernel='rbf'`, `C=1`, `epsilon=0.001`

For the SVR, the default parameters provided by *sklearn* are used, since these provided a better performance than the parameters found by random search.

D. Model Performance

To validate the classifiers, 5-fold Cross Validation is used. The results are presented in Table I. The *Mean Squared Error (MSE)*, *Mean Absolute Error (MAE)*, and *R2* serve as common regression metrics to discuss the performance of the classifiers. For all three link metrics, the random forest model achieved the best performance across all regression metrics. The RF had a better MAE by 10 Mbit/s compared to the second-best model, the SVR. The R2 is higher than 0.3 for each metric, for download throughput even higher than 0.4, meaning that 40% of the variance in download throughput can be explained by the weather features. We expected such a good results for the download throughput, since the authors of WetLinks dataset paper [3] discussed a high impact of rain on the download throughput. The results obtained by all three classifiers for upload throughput and RTT are quite surprising, as they reveal an almost non-existent correlation, which contradicts the analysis provided in the dataset paper. However, in combination of the variety of features, especially using future and past values, the RF model was able to provide rather good predictions for upload throughput and RTT as well. Compared to the RF, the two other models (MLP and SVR), show worse performance. The SVR is capable of achieving slightly better results than the MLP. The classifiers for these two models perform better than the dummy classifier in the download throughput case, however, the MLP and SVR classifiers cannot challenge the Dummy classifier in the upload throughput and RTT prediction. In conclusion, the Random Forest achieves the best results in the evaluation for predicting RTT, download, and upload throughput. We assume that the architecture of a random forest is the best capable of building a regressor that generalizes properly to these use-cases and detects the patterns

¹<https://scikit-learn.org/stable/>

of the weather influence on the link. Due to its architecture, the RF can provide explanations based on feature importance. The examination of these importance yields valuable insights into the generalization capability of a classifier. For example, an examination of the download throughput classifier yields the following results. The most important feature is *rain_past_5p* (5min past window) with a Feature importance of 0.027. The second and third important features are *rain_future_10p* with 0.023 and *windspeed_past_5p* with 0.022. Since a rain window in this dataset is by median 12 minutes long, we argue that the feature importance of the rain past and rain future features are reasonable. However, the importance of wind and wind direction, which also appears more often in the latter important places alternating with rain features, is not obvious to us. One assumption could be that, the classifier differentiates, in which direction and speed clouds are pushed by the wind, to estimate when a cloudy condition comes in or the sky clears up. Nevertheless, more investigation is needed on this.

III. DEMONSTRATION SETUP

The demo setup shown to the audience visualizes the performance of the three pre-trained AI models. During the demo presentation, predictions of round-trip time, download throughput, and upload throughput will be generated based on weather data using a computer equipped with pre-trained models. These predictions will appear on the dashboard dynamically, along with the weather data used to make the predictions. A screenshot of the dashboard can be found in Figure 1. On the left side of the dashboard, weather conditions from the past 24 hours and the 10-minute forecasts are displayed. Four graphs display solar radiation, humidity, precipitation, and wind speed. Additionally, a current image of the sky along with an icon is displayed to provide a visual impression of the current weather conditions. On the right side, forecasts for RTT, download throughput, and upload throughput are presented alongside actual measured values, providing a comparison between prediction and reality, as well as evaluating the quality of different models. The screenshot captures a moment at which the download and upload throughputs decrease due to weather influence. This decrease is predicted by all three AI models, demonstrating both the influence of weather on link quality and the ability to forecast its effects by different models. By providing a holistic view of current weather conditions alongside forecasts for network metrics, the demo setup effectively visualizes the correlation between weather conditions and link metrics in satellite communications. This insight is valuable for proactively optimizing satellite communication systems, ensuring reliable connectivity and maximizing performance under varying environmental conditions.

IV. DISCUSSION & FUTURE WORK

This demo visualizes the feasibility of predicting link parameters based solely on historical and forecast weather data. Among our models, Random Forest emerged as the most effective. Despite its good performance, our analysis revealed

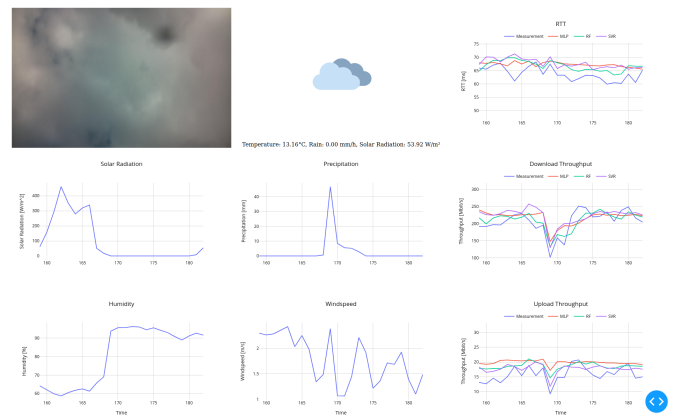


Fig. 1: Dashboard that will be presented to the audience.

that error rates remain relatively high, with an MAE of 36 Mbit/s for the Download throughput. While this indicates that factors other than weather conditions also affect link parameters, we demonstrate that the performance is sufficient to predict general trends. Examining the feature importance of RF, we found that the model heavily relies on rain and wind features for prediction. Notably, the relationship between wind and link parameters appears less clear-cut. For future work, we plan to further explore the impact of weather on the RTT. In addition, we aim to explore more sophisticated models tailored for time series data, such as Long Short-Term Memory (LSTM) networks. Furthermore, we intend to investigate diverse feature reduction pipelines in order to streamline our feature space. Moreover, we plan to conduct an investigation into classifiers, particularly focusing on features related to wind. Finally, we will conduct an analysis to identify and understand other potential influences on link parameters beyond weather conditions.

V. ACKNOWLEDGMENT

This work has been partially supported by the German Federal Ministry for Digital and Transport as part of the “Innovative Network Technologies” funding program (FKZ: 19OI23008C).

REFERENCES

- [1] L. Izhikevich, M. Tran, K. Izhikevich, G. Akiwate, and Z. Durumeric, “Democratizing LEO Satellite Network Measurement,” *Proc. of the ACM on Measurement and Analysis of Computing Systems*, vol. 8, no. 1, 2024.
- [2] M. M. Kassem, A. Raman, D. Perino, and N. Sastry, “A Browser-side View of Starlink Connectivity,” in *Proc. of the 22nd ACM Internet Measurement Conf. (IMC ’22)*, 2022, p. 151–158.
- [3] D. Laniewski, E. Lanfer, B. Meijerink, R. van Rijswijk-Deij, and N. Aschenbruck, “WetLinks: a Large-Scale Longitudinal Starlink Dataset with Contiguous Weather Data,” in *Proc. of the 8th Network Traffic Measurement and Analysis Conf. (TMA)*, 2024.
- [4] S. Ma, Y. C. Chou, H. Zhao, L. Chen, X. Ma, and J. Liu, “Network Characteristics of LEO Satellite Constellations: A Starlink-Based Measurement from End Users,” in *Proceedings of the IEEE International Conf. on Computer Communications (INFOCOM)*, 2023, pp. 1–10.
- [5] J. McDowell, “Enormous (‘Mega’) Satellite Constellations,” <https://planet4589.org/space/con/conlist.html>, 2024.
- [6] F. Michel, M. Trevisan, D. Giordano, and O. Bonaventure, “A First Look at Starlink Performance,” in *Proc. of the 22nd ACM Internet Measurement Conf. (IMC ’22)*, 2022, p. 130–136.