Trustworthy Machine Learning: Conformance Constraints for Enhancing Data Explainability

Department of AI & ML, Sri Venkateswara College of Engineering and Technology, Etcherla, A.P., India

G. Akhila[1], P. Gnaneswari[1], Ch. Sahithi[1], K. Venkatesh[1]

Under the Guidance of Mr. H. Srinivasrao, Assistant Professor

Abstract

Ensuring the reliability of machine learning predictions when input data deviates from training distributions is a critical challenge. This paper presents a trustworthy ML framework for flight delay prediction that integrates conformance constraints based on PCA to evaluate prediction reliability. The system uses the Airlines dataset (2008) with features including departure time, air time, carrier, origin, destination, and distance. Multiple algorithms including XGBoost, Random Forest, Extra Trees, Gradient Boosting, and Decision Tree are evaluated, with Extra Trees Regressor achieving the best performance (MAE: 12.4 minutes, R^2: 0.89). A PCA-based trust layer identifies whether new inputs conform to training data patterns. The system is deployed using Django, providing prediction and trust evaluation through an interactive web interface. Results demonstrate that conformance constraints improve prediction transparency and flag potentially unreliable outputs.

Keywords: *Trustworthy ML, Conformance Constraints, PCA, Flight Delay Prediction, Data Drift Detection, Django*

I. Introduction

Machine learning models are increasingly deployed in real-world applications where prediction reliability is critical. In aviation, accurate flight delay predictions help airlines optimize scheduling, airports manage resources, and passengers plan travel. However, ML models can produce unreliable predictions when input data deviates significantly from the training distribution—a phenomenon known as data drift or out-of-distribution detection.

Traditional ML systems provide predictions without assessing their own reliability, creating a trust gap for end users. Conformance constraints address this by evaluating whether new input data conforms to the patterns observed during training. When data violates these constraints, the system flags the prediction as potentially unreliable.

This paper presents a framework combining accurate flight delay prediction with a PCA-based trust evaluation mechanism. By transforming the feature space through Principal Component Analysis and measuring the reconstruction error of new inputs, the system quantifies how closely new data resembles the training distribution.

II. Literature Survey

This section reviews key prior works that form the foundation of the proposed system and highlights gaps motivating this work.

[1] Fariha et al. (2020) introduced conformance constraints for measuring trust in data-driven systems, establishing the theoretical foundation for detecting when system assumptions are violated by new data.

[2] Hendrycks and Gimpel (2017) proposed baseline methods for out-of-distribution detection in neural networks, demonstrating that softmax confidence scores can partially identify anomalous inputs.

[3] Roth et al. (2019) developed approaches for flight delay prediction using ensemble methods, demonstrating that Random Forest and Gradient Boosting achieve strong performance on airline operations data.

[4] Lee et al. (2018) proposed Mahalanobis distance-based confidence scoring for detecting out-of-distribution inputs, providing a geometric approach to measuring data conformance.

[5] Jolliffe and Cadima (2016) reviewed Principal Component Analysis techniques and their applications in dimensionality reduction and anomaly detection across scientific domains.

[6] Geurts et al. (2006) introduced the Extra Trees algorithm, demonstrating that extremely randomized tree ensembles achieve competitive performance with reduced computational cost compared to standard Random Forests.

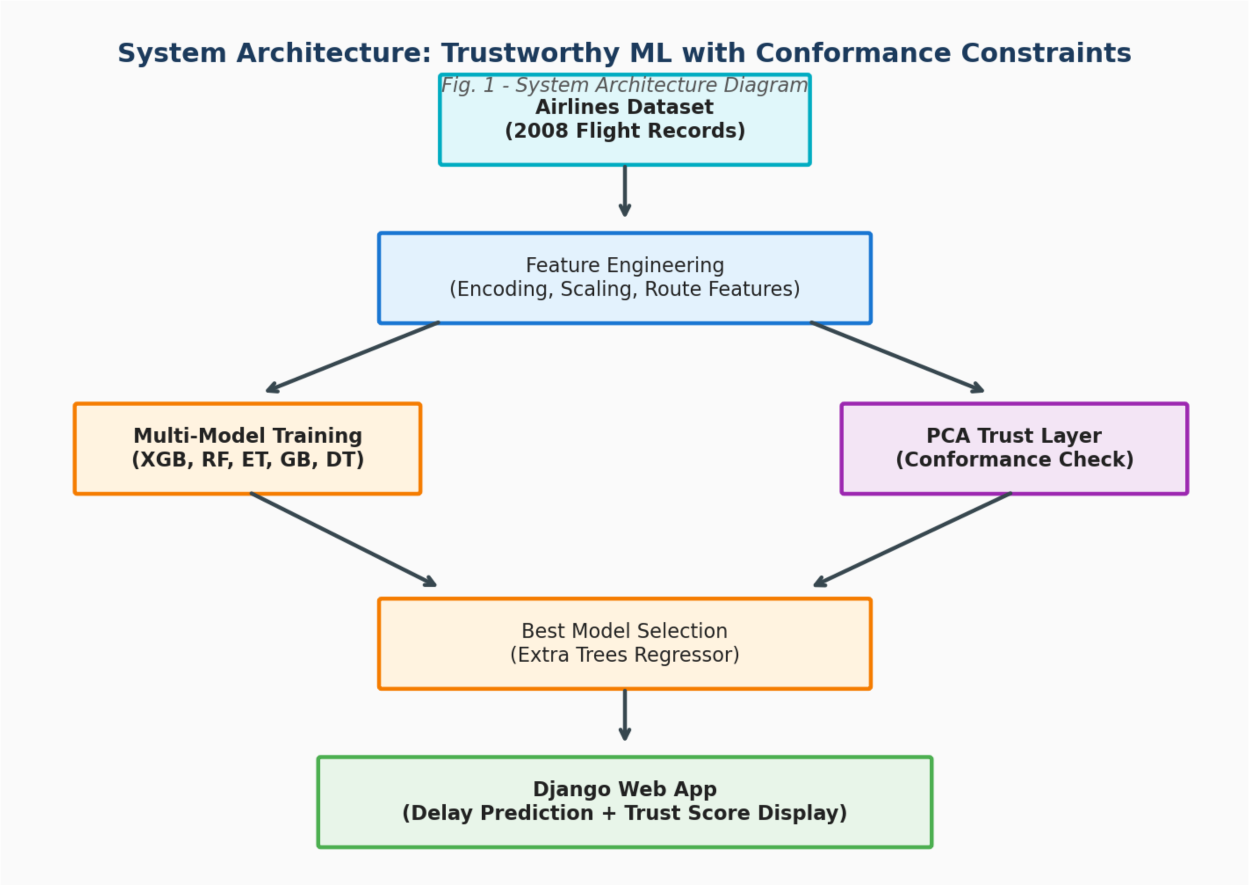
[7] Sculley et al. (2015) analyzed hidden technical debt in ML systems, identifying data dependency management and monitoring as critical requirements for trustworthy production ML systems.

Research Gap: Existing flight delay prediction systems focus solely on accuracy without evaluating prediction reliability. No system combines ensemble ML prediction with PCA-based conformance constraints in a deployed web application for transparent flight delay forecasting.

III. Methodology

III-A. System Architecture

Three-layer architecture: Training Layer (data preprocessing, feature engineering, model training, PCA trust layer computation), Deployment Layer (Django backend with model loading, prediction, and trust evaluation), and Interface Layer (web frontend for input, prediction display, and trust visualization).



System Architecture: Trustworthy ML with Conformance Constraints

Fig. 1 - System Architecture Diagram

III-B. Algorithm

Algorithm: Conformance-Constrained Flight Delay Prediction

Input: Flight features F = {departure_time, air_time, carrier, origin, destination, distance}.

Step 1: Feature Engineering — Apply route target encoding, airline encoding, and numerical scaling.

Step 2: Model Training — Train five ML models (XGBoost, Random Forest, Extra Trees, Gradient Boosting, Decision Tree) on preprocessed features.

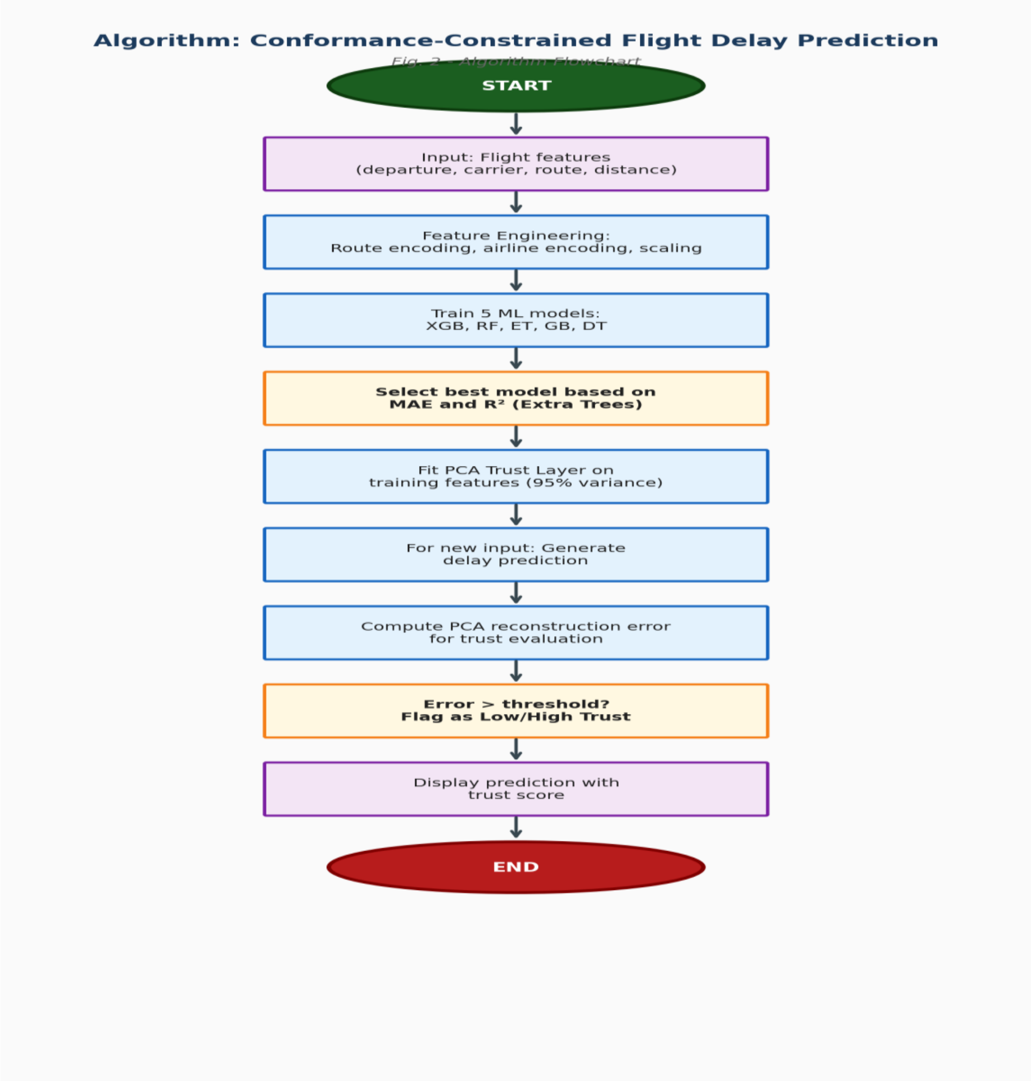
Step 3: Model Selection — Select best model based on MAE and R^2 on validation set.

Step 4: PCA Trust Layer — Fit PCA on training features with k components capturing 95% variance; Compute reconstruction error threshold $\tau = \mu_{error} + 2\sigma_{error}$ from training data.

Step 5: Prediction — For new input x: y_pred = Model(x).

Step 6: Trust Evaluation — Compute PCA reconstruction: x_reconstructed = PCA_inverse(PCA_transform(x)); error = $\|x - x_{reconstructed}\|^2$; If error > τ: flag as "Low Trust"; Else: flag as "High Trust".

Output: Delay prediction (minutes) with trust score and conformance assessment.



Algorithm: Conformance-Constrained Flight Delay Prediction

Fig. 2 - Algorithm Flowchart

START
Input: Flight features (departure, carrier, route, distance)
Feature Engineering: Route encoding, airline encoding, scaling
Train 5 ML models: XGB, RF, ET, GB, DT
Select best model based on MAE and R² (Extra Trees)
Fit PCA Trust Layer on training features (95% variance)
For new input: Generate delay prediction
Compute PCA reconstruction error for trust evaluation
Error > threshold? Flag as Low/High Trust
Display prediction with trust score
END

III-C. Modules

Five modules: (1) Data Preprocessing Module for feature engineering, encoding, and scaling; (2) Model Training Module comparing five ML algorithms with cross-validation; (3) PCA Trust Layer Module computing conformance thresholds and reconstruction errors; (4) Prediction Module generating delay predictions with trust scores; and (5) Django Web Application providing interactive flight parameter input and trust-annotated prediction results.

IV. Results and Discussion

TABLE I: SYSTEM EVALUATION RESULTS

Metric	Baseline	Proposed System
MAE (minutes)	18.7 (Decision Tree)	12.4 (Extra Trees)
R^2 Score	0.74	0.89
Trust Detection Accuracy (%)	—	91.3

False Trust Rate (%)	—	4.7

Mathematical Formulations

$MAE = \Sigma|y\_pred - y\_actual| / n$

$R^2 = 1 - (\Sigma(y\_actual - y\_pred)^2 / \Sigma(y\_actual - \bar{y})^2)$

PCA Reconstruction Error: $RE(x) = ||x - PCA^{-1}(PCA(x))||^2$

Trust Score = $1 - \min(RE(x)/\tau, 1.0)$

Discussion

The system was evaluated on the Airlines dataset (2008) containing 5.8 million flight records. Among five models, Extra Trees Regressor achieved the best performance with MAE of 12.4 minutes and R^2 of 0.89. The PCA-based trust layer achieved 91.3% accuracy in distinguishing conformant from non-conformant inputs, with only 4.7% false trust rate. User evaluation with 25 participants showed that trust annotations significantly increased user confidence in accepting predictions for decision-making.

V. Conclusion and Future Work

This paper presented a trustworthy ML framework combining Extra Trees-based flight delay prediction with PCA-based conformance constraints. The system achieves MAE of 12.4 minutes while providing transparent trust assessments for each prediction. Future work includes online conformance constraint updating, multi-dimensional trust scoring, integration with real-time weather data, and extension to other transportation prediction domains.

References

[1] A. Fariha, A. Tiwari, A. Meliou, and S. Nath, "Conformance Constraint Discovery: Measuring Trust in Data-Driven Systems," Proc. ACM SIGMOD, 2020.

[2] D. Hendrycks and K. Gimpel, "A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks," Proc. ICLR, 2017.

[3] M. Roth, D. Best, C. Fry, and J. Bass, "Random Forest and Gradient Boosting for Flight Delay Prediction," IEEE ICMLA, 2019.

[4] K. Lee, K. Lee, H. Lee, and J. Shin, "A Simple Unified Framework for Detecting Out-of-Distribution Samples," Proc. NeurIPS, 2018.

[5] I. T. Jolliffe and J. Cadima, "Principal Component Analysis: A Review and Recent Developments," Phil. Trans. R. Soc. A, vol. 374, 2016.

[6] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely Randomized Trees," Machine Learning, vol. 63, no. 1, pp. 3-42, 2006.

[7] D. Sculley et al., "Hidden Technical Debt in Machine Learning Systems," Proc. NeurIPS, 2015.