Journal of Electrical Engineering and Computer (JEECOM)
Vol. 8, No. 2 (206), DOI: 10.33650/jeecom.v8i2.17161
p-ISSN: 2715-0410; e-ISSN: 2715-6427

From Black Box to Grid-Ready: Explainable CRISP-DM for High-Demand EV Charging Prediction

Dwi Putro Sarwo Setyohadi [1*], Hendra Yufit Riskiawan [2], Akas Bagus Setiawan [3]
[1,2,3] Department of Information Technology, Politeknik Negeri Jember, Jember, Indonesia

Article Info

Article history:

Received August 29, 2026
Revised September 23, 2026
Accepted October 10, 2026

Keywords:

Class Imbalance
CRISP-DM
Electric Vehicle Charging
Explainable AI
SMOTE
Streamlit
XGBoost

ABSTRACT

The rapid growth of electric vehicle (EV) adoption creates uncertainty for power grid operators in anticipating charging load, particularly sessions that draw a disproportionately large amount of energy. This study applies the CRISP-DM (Cross-Industry Standard Process for Data Mining) framework to a large-scale synthetic EV charging dataset (1,965,239 sessions; four raw attributes: connection time, charging duration, energy delivered, and a sequential day index) to build an early, context-only classifier for *high-demand* sessions, defined as sessions whose delivered energy falls in the top quartile (≥75th percentile). To avoid label leakage, only information available at the start of a session connection time, planned duration, a peak-hour flag, and a weekend flag was used as predictors. The resulting target is naturally imbalanced (75:25); its effect was investigated through an ablation study comparing five classification algorithms (Logistic Regression, Decision Tree, Random Forest, XGBoost, and K-Nearest Neighbors) with and without SMOTE oversampling, evaluated on accuracy, precision, recall, F1-score, and ROC-AUC. The best configuration, XGBoost combined with SMOTE, achieved F1-score = 0.600 and ROC-AUC = 0.818, improving recall from 0.442 to 0.807 at a moderate precision cost. SHAP-based attribution identified charging duration as the dominant predictor in the trained model, followed by connection time, consistent with the raw feature correlations (r = 0.51 and r = 0.30, respectively). The full pipeline data preparation, modeling, ablation, and explainability were deployed as an interactive, open web application built with Streamlit, allowing non-technical stakeholders to reproduce the analysis and inspect results in real time.

Corresponding Author:
Dwi Putro Sarwo Setyohadi
Department of Information Technology, Politeknik Negeri Jember, Jember, 68121, Indonesia
Email: dwi.putro@polije.ac.id; Orcid ID : https://orcid.org/0000-0002-5211-9738

1. INTRODUCTION

The global shift toward electric vehicles (EVs) is accelerating, and with it comes a new source of stochastic, potentially large electrical load on distribution networks. Utilities and charging-network operators increasingly need planning tools that go beyond aggregate forecasts and can flag, session by session, which connections are likely to draw substantially more energy than average, information that directly supports hosting-capacity studies, tariff design, and charger-sizing decisions [1], [2], [3]. Dynamic, demand-responsive tariffs in particular rely on being able to anticipate high-load sessions early, since load-shifting incentives are only effective if they are applied before, not after, a large session begins [4]. Because EV charging behavior is

highly heterogeneous, shaped by driver habits, vehicle battery size, time of arrival, and local pricing schemes, this remains a non-trivial forecasting and classification problem [5].

In response to this challenge, a growing body of applied machine-learning research has begun to address EV charging prediction with explainable models so that grid operators can inspect and act on individual predictions rather than treating the model as an opaque black box [6], [7], [8]. However, three issues are still rarely handled together in one reproducible workflow: high-demand sessions are minority events, model comparisons often do not isolate the specific contribution of imbalance correction, and explanation results are frequently reported without being embedded in a deployable decision-support artifact. These limitations define the research gap addressed in this study and motivate the detailed review in Section 2.

This study addresses that gap by applying CRISP-DM end-to-end to a large synthetic EV charging dataset (≈1.97 million sessions) to classify *high-demand* charging sessions using only information available at session start. The specific contributions of this paper are: (1) a leakage-free feature/target design in which the classification target is derived from delivered energy while the predictors are restricted to context available at connection time; (2) a controlled ablation study comparing five classification algorithms with and without SMOTE, isolating the effect of class-imbalance correction from the effect of algorithm choice; (3) an explainability layer combining SHAP and permutation importance to examine whether model behavior is consistent with domain intuition; and (4) an open-source, interactive Streamlit web application that reproduces every stage of the pipeline and automatically archives all resulting tables and figures as research artifacts. The remainder of this paper is organized as follows: Section 2 situates these contributions within related work on EV demand prediction, class imbalance, explainability, and reproducible pipelines; Section 3 details the CRISP-DM method, including the formal definitions of the target, resampling procedure, and each classifier; Section 4 reports and discusses the ablation and explainability results; and Section 5 concludes with limitations and future work.

2. RELATED STUDIES

2.1. EV Charging Demand Prediction and Grid Impact

Grid-side interest in EV charging prediction is driven by the concrete engineering problem of hosting capacity: how much additional EV load a given distribution network can absorb before violating voltage or thermal limits [1], [3]. Reviews of hosting-capacity quantification methods converge on the same conclusion: accurate, session-level estimates of *how much* energy a connection will draw are more actionable for planners than coarse, aggregate load forecasts [3]. On the demand-response side, dynamic and time-of-use tariff schemes have been shown to shift load away from peak periods, but only when high-load sessions can be flagged before, rather than after, they occur [4]. At the forecasting-methods level, comparative studies evaluating statistical, machine-learning, and deep-learning approaches on real EV charging-session data consistently report that no single family of models dominates across all time horizons, motivating careful, task-specific model selection rather than defaulting to the most complex available architecture [5].

2.2. Machine Learning and Explainability for EV Charging Behavior

Beyond aggregate load forecasting, several studies have modeled charging behavior at the level of the individual session or driver decision. Tree-ensemble classifiers, and XGBoost in particular, have repeatedly been reported as the top performer when predicting discrete charging-related outcomes, including which charging station a driver will choose [6] and hourly charging demand in renewable-powered grids [7] when compared against simpler baselines such as logistic regression or single decision trees. A recurring design choice in this literature is to pair the predictive model with SHAP-based explanation, which lets researchers verify that the learned decision rules are consistent with known driver behavior (e.g., dwell time, price sensitivity) rather than being artifacts of the training data. This same explainability requirement extends beyond the charging-session level to AI governance in smart energy systems more broadly, where explainability is repeatedly identified as a necessary but still under-operationalized condition for stakeholder trust in automated grid decision-making [8].

2.3. Handling Class Imbalance in Predictive Modeling

Trustworthy explanations are of limited use, however, if the underlying classifier has learned to ignore the event of interest in the first place, a risk that is especially acute in EV charging data. Outcomes of practical interest in EV charging a session becoming unusually long, unusually energy-intensive, or ending in overstay are, by construction, minority events. Broad surveys of imbalanced learning confirm that classifiers trained without correction systematically default toward the majority class, inflating accuracy while suppressing recall

for the event that actually matters operationally [9]. SMOTE (Synthetic Minority Over-sampling Technique) and its many subsequent variants remain the most widely adopted family of corrections; refinements such as noise-filtering before or after synthetic-sample generation have been shown to further improve classifier recall on imbalanced tabular data without a proportional loss of precision [10]. This study follows the same data-level correction philosophy, resampling the training data rather than modifying the learning algorithm to keep the comparison across five different classifiers straightforward and algorithm-agnostic.

2.4. Explainable Artificial Intelligence (XAI)

Correcting for imbalance, in turn, changes *how* a model reaches its decisions, which makes explaining those decisions afterward even more important. As predictive models move from linear baselines to tree ensembles, the resulting gain in accuracy is frequently offset by a loss of transparency, which is problematic in operational settings where a human planner must act on a model's output. Systematic reviews of the XAI literature categorize explanation methods along two axes, model-specific versus model-agnostic, and intrinsic versus post-hoc, and report that post-hoc, game-theoretic methods such as SHAP have become the de facto standard for explaining tree-ensemble predictions because they provide both global (feature-importance) and local (per-prediction) explanations from a single, theoretically grounded framework [11]. Practical guidance on applying SHAP in applied, non-machine-learning-specialist settings further recommends pairing the global summary plot used in this study with model-agnostic sanity checks, such as permutation importance, before drawing domain conclusions from feature attributions [12].

2.5. CRISP-DM, MLOps, and Reproducible Deployment

None of the practices reviewed above imbalance correction, model selection, and explanation are useful in isolation; they must be assembled into a single, auditable pipeline that others can run and trust. CRISP-DM (Cross-Industry Standard Process for Data Mining) remains the most widely referenced process model for structuring applied data-mining projects into auditable phases: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. Case studies applying CRISP-DM outside idealized, textbook settings report that the generic six-phase model typically requires domain-specific adaptation, for instance, adding explicit regulatory-compliance and governance checkpoints in regulated industries while still preserving the core phase structure [13]. Once a pipeline moves past a single exploratory run, class-imbalance correction itself must be treated as a first-class, versioned pipeline component rather than an ad-hoc script; open-source toolboxes that expose resampling methods through the same interface as standard classifiers materially reduce this integration burden [14]. Related applications in social-media risk screening and clinical-operational prediction further show how imbalance handling, explainability, and operational decision support can be integrated into applied pipelines beyond the EV domain [34], [35]. More broadly, the MLOps literature covering automation, versioning, monitoring, and reproducibility of ML systems in production has matured from scattered tool support into systematic maturity models, offering concrete practices for exactly the kind of artifact-archiving, re-runnable pipeline built in this study [15], [16].

Taken together, this body of work establishes that (a) session-level, tree-ensemble models are effective for EV charging prediction tasks, (b) SMOTE-family correction is the standard response to the class imbalance inherent in such tasks, (c) SHAP-based explanation is the standard companion to tree-ensemble predictions in operational settings, and (d) CRISP-DM and MLOps practices provide the process scaffolding for making such pipelines auditable and reproducible. What remains comparatively rare is a *single* pipeline that combines all four elements: a standard, auditable process model; an explicit ablation design that isolates the marginal effect of imbalance correction from algorithm choice, rather than assuming SMOTE helps uniformly; a systematic explainability layer; and a packaged, interactive artifact that a grid planner or reviewer can operate directly, instead of a static notebook. This study addresses that gap, operationalized in Section 3 as a six-phase CRISP-DM pipeline.

3. METHOD

Guided by the gap identified above, this study follows the CRISP-DM process model [13], comprising Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. Each phase below corresponds to one module of the accompanying software pipeline, beginning with the business question that motivates every subsequent design choice. The research flow is summarized in Figure 1.

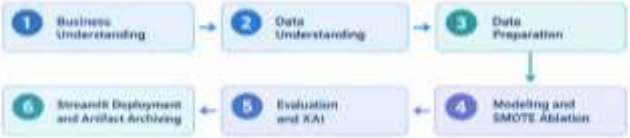

Figure 1. Research flow diagram for the CRISP-DM pipeline.

ISSN: 2715-6427

3.1. Business Understanding

The business question is: *given only the context available when a charging session begins (arrival time, planned duration, whether it falls in a peak hour, and whether it falls on a weekend), can we predict early whether the session will end up delivering a disproportionately large amount of energy ("high demand")?* An accurate early classifier supports grid-capacity planning and demand-side management, since operators can pre-emptively allocate capacity or apply dynamic pricing to sessions flagged as high-risk for high demand [1], [2].

3.2. Data Understanding

Using a synthetic rather than a directly identifiable dataset sidesteps the privacy-measurement and disclosure-risk concerns that accompany real transaction-level logs concerns increasingly formalized in regulatory guidance for the release of synthetic tabular data [17] while still allowing the full CRISP-DM pipeline to be demonstrated end-to-end; high-resolution, multidimensional real-world charging transaction datasets have only recently begun to be released publicly [18], reinforcing the case for first validating the pipeline design on a synthetic proxy. The dataset consists of 1,965,239 synthetic EV charging sessions with four raw attributes: connectionTime_decimal (hour of day, 0–24, at which the vehicle connects), chargingDuration (session length), kWhDelivered (energy delivered), and dayIndicator (a sequential session/day index, range 1–29,600). Exploratory analysis in Figure 2 shows dayIndicator is essentially uncorrelated with the other three variables ($r \approx 0.00$–0.01), indicating the dataset was generated independently of any explicit weekly pattern; by contrast, chargingDuration correlates moderately with kWhDelivered ($r = 0.51$), and connectionTime_decimal correlates weakly with kWhDelivered ($r = 0.30$). This finding directly informed the target design in Section 3.3.

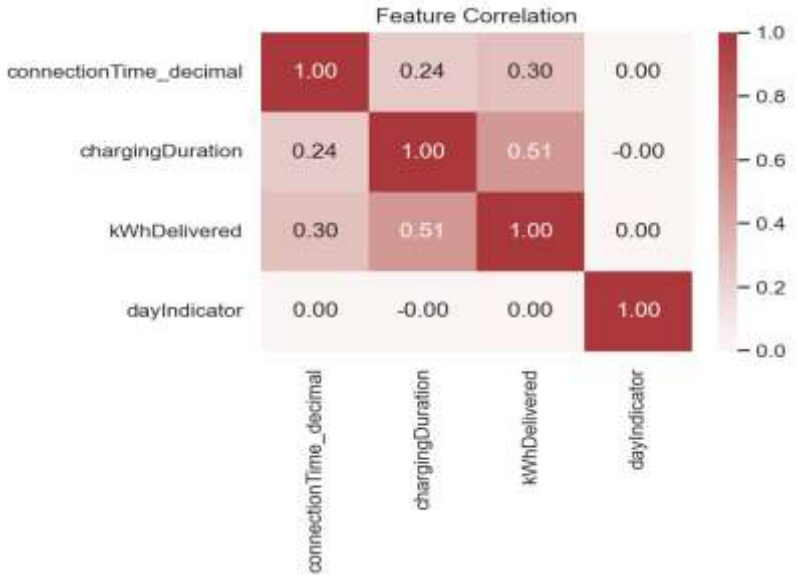

Figure 2. Correlation heatmap of raw session attributes.

For computational tractability inside an interactive web application, a stratified random subsample (n = 60,000, seed = 42) was drawn from the full dataset using chunked reading (200,000 rows per chunk) to bound memory use; a further configurable subsample (default n = 20,000) is drawn at runtime for model training, while the full-resolution file remains available for offline/local runs.

3.3. Data Preparation

Because dayIndicator carries no measurable behavioral signal (Section 3.2), it was not used to define the prediction target. Instead, the binary target high_demand was defined by thresholding delivered energy at its 75th percentile, computed over the working sample:

$$y_i = \mathbb{1}[\,\text{kWhDelivered}_i \geq Q_{75}(\text{kWhDelivered})\,] \quad (1)$$

where $\mathbb{1}[\cdot]$ is the indicator function and $Q_{75}(\cdot)$ denotes the 75th-percentile (upper-quartile) operator. kWhDelivered itself is excluded from the feature set to avoid label leakage, a failure mode identified as a leading, often silent cause of overoptimistic and irreproducible results across machine-learning-based science [19].

Feature engineering: Four context features, all computable at or before session start, were derived: connectionTime_decimal, chargingDuration, is_peak_hour (1 if connection time falls between 17:00–21:00), and is_weekend (1 if the session's position in the dayIndicator cycle falls on the 6th or 7th day of a 7-day cycle). Rows with missing values or physically invalid readings (negative duration/energy, connection time outside [0, 24)) were removed.

Class-imbalance handling: The resulting target distribution is imbalanced (≈75% Normal/Low-Demand vs. ≈25% High-Demand), consistent with imbalance patterns widely reported for rare-event classification tasks [9]. SMOTE (Synthetic Minority Over-sampling Technique) and its noise-aware variants [10], implemented via the imbalanced-learn toolbox [12], were applied exclusively to the training split. For a minority-class instance x_i, a synthetic sample is generated as

$$x_{\text{new}} = x_i + \lambda \cdot (x_{\text{zi}} - x_i), \qquad \lambda \sim U(0,1) \tag{2}$$

where x_{zi} is one of the k nearest minority-class neighbors of x_i and λ is a random interpolation factor drawn uniformly from $[0,1]$, i.e., x_{new} lies on the line segment between x_i and x_{zi}. This procedure was repeated until the minority class matched the majority class in the training split; the test split was left untouched to prevent optimistic bias.

3.4. Modeling

Consistent with survey evidence that tree-based ensembles continue to match or outperform deep learning on tabular data [20], five supervised classification algorithms were selected to represent complementary inductive biases suited to a low-dimensional, numeric, tabular classification problem.

Logistic Regression [21] models the log-odds of the positive class as a linear function of the input features, mapped to a probability through the sigmoid function:

$$P(y = 1 \mid x) = \frac{1}{1+e^{-(w^{\mathrm{T}}x+b)}} \tag{3}$$

where w is the learned weight vector and b the bias term.

Decision Tree [22], following the CART/C4.5 family of recursive-partitioning classifiers, greedily selects, at each node t, the split that most reduces class impurity, measured here by the Gini index:

$$Gini(t) = 1 - \sum_{k=1}^{K} p_k^2 \tag{4}$$

where p_k is the proportion of class-k samples at node t and K is the number of classes.

Random Forest [23] is a bagging ensemble of B decision trees, each trained on a bootstrap sample with a random feature subset; the ensemble prediction is the majority vote across trees:

$$\hat{y} = \text{mode}\{h_1(x), h_2(x), \ldots, h_B(x)\} \tag{5}$$

which reduces variance relative to a single tree and provides built-in feature-importance and permutation-importance diagnostics [23].

XGBoost [24] is a gradient-boosted tree ensemble that sequentially fits new trees f_k to the residual error of the current ensemble, minimizing a regularized objective:

$$\mathcal{L}(\phi) = \sum_i l\,(y_i, \hat{y}_i) + \sum_k \Omega\,(f_k) \tag{6}$$

where $l(\cdot)$ is a differentiable loss (log-loss for binary classification) and $\Omega(f_k)$ penalizes tree complexity to control overfitting.

K-Nearest Neighbors [25] is a non-parametric, distance-based classifier that assigns the majority label among the k closest training instances, using Euclidean distance:

$$d(x_i, x_j) = \sqrt{\sum_{m=1}^{p} (x_{im} - x_{jm})^2} \tag{7}$$

Logistic Regression and K-Nearest Neighbors, being scale-sensitive, were trained on standardized features (zero mean, unit variance); the three tree-based models were trained on the untransformed features.

Ablation design. Each of the five algorithms was trained under two conditions *No-SMOTE* and *SMOTE,* yielding 10 model runs in total, all sharing an identical 75/25 stratified train–test split (seed = 42). This design isolates the marginal effect of SMOTE from the effect of algorithm choice, rather than assuming a single "best" combination a priori, following the logic of factorial experimental design, in which each factor's contribution is isolated by varying one condition at a time while holding the others fixed [26].

ISSN: 2715-6427

Evaluation metrics**.** Each run was evaluated on the untouched test split using:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

$$Precision = \frac{TP}{TP+FP}, \qquad Recall = \frac{TP}{TP+FN} \quad (9)$$

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall} \quad (10)$$

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives with respect to the high_demand class. ROC-AUC (area under the receiver operating characteristic curve) was computed as a threshold-independent summary of ranking quality. All metrics were computed with scikit-learn [27].

3.5 Evaluation and Explainable AI (XAI)

Beyond the quantitative metrics above, model behavior was interpreted using SHAP (SHapley Additive exPlanations) [12], which attributes each prediction to individual feature contributions grounded in cooperative game theory. For a feature i and prediction function f, the Shapley value is

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!\,(|F|-|S|-1)!}{|F|!} [f(S \cup \{i\}) - f(S)] \quad (11)$$

where F is the full feature set and S ranges over all feature subsets excluding i i.e., ϕ_i is the average marginal contribution of feature i across every possible ordering of features. For tree-based models, the exact TreeExplainer variant [28] was used; for other models, a sample-based KernelExplainer was used. As a complementary, model-agnostic check, permutation feature importance [23] was computed for non-tree models by measuring the drop in performance when a feature's values are randomly shuffled.

3.6 Deployment

All stages above were implemented as a modular Python pipeline (data.py, modeling.py, viz.py, xai.py) following MLOps principles of automation, versioning, and reproducibility [15], [16], consistent with the pipeline architecture used in the author's prior work on end-to-end ML deployment [29]. The pipeline is wrapped in an interactive web application built with Streamlit [30], consistent with the broader trend of embedding machine-learning-based forecasting and decision-support directly into operational tooling for grid-connected energy systems [31]. Figure 3 illustrates the application landing page, where the seven tabs mirror the CRISP-DM phases and the sidebar exposes the run-time configuration controls. Every table (CSV) and figure (PNG) produced during a pipeline run class-balance plots, correlation heatmap, ablation comparison, confusion matrices, ROC curves, feature importance, and SHAP summary is automatically archived to a dedicated asset directory as a reproducible research artifact. Plotting was performed with Matplotlib [32], and pandas [33] was used throughout for data handling. The application accepts either the bundled representative sample or a user-uploaded CSV, and is designed for direct deployment on Streamlit Community Cloud without requiring the full 115 MB source dataset. With the pipeline and its deployment mechanism now fully specified, Section 4 walks through its output at each of these stages, in the same order in which they execute.

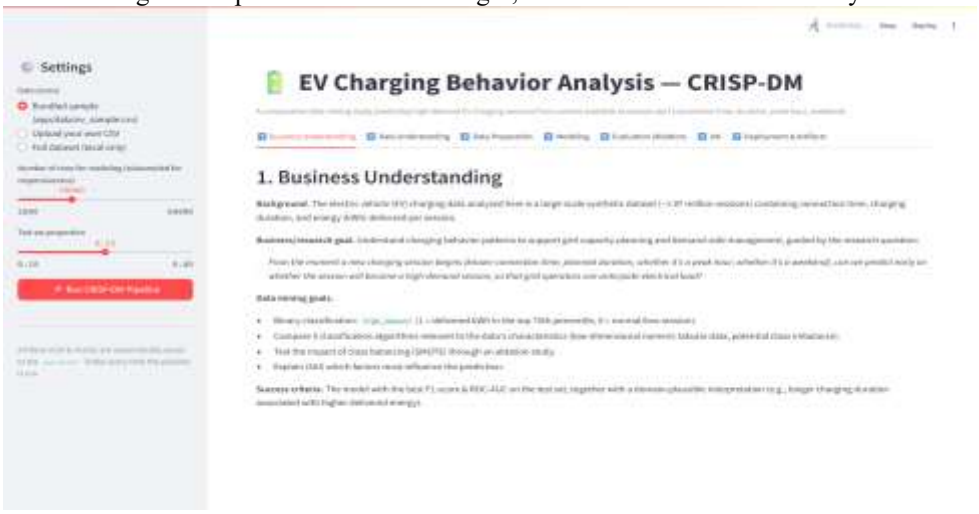

Figure 3. Streamlit application landing page, showing the seven CRISP-DM.

4. RESULTS AND DISCUSSION

4.1. Class Imbalance and the Effect of SMOTE

Execution begins where Data Preparation left off: as anticipated in Section 3.3, the high_demand target is imbalanced: 75.0% Normal/Low-Demand vs. 25.0% High-Demand in the working sample (Figure 4). After SMOTE was applied to the training split only, the training-set distribution was rebalanced to an approximately 1:1 ratio (Figure 5), while the held-out test split retained its original, real-world proportions, ensuring that the evaluation metrics reported below reflect performance on realistic, non-synthetic class proportions.

Figure 4. Class distribution of the high_demand target before SMOTE.

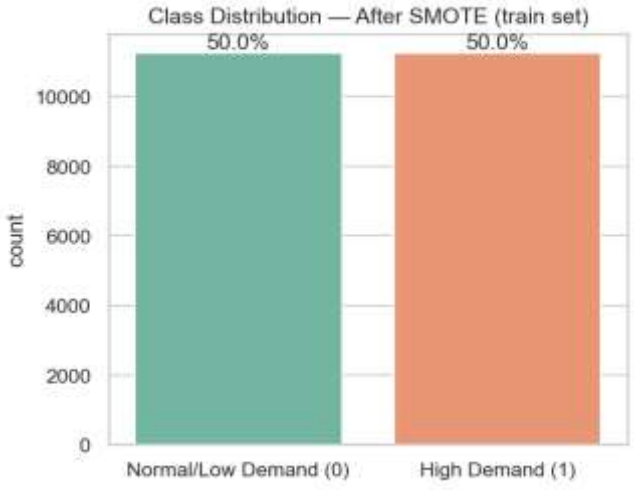

Figure 5. Class distribution of the training split after SMOTE.

4.2. Model Comparison and Ablation Study

Table 1 reports the full ablation results across all five algorithms and both conditions (n = 20,000 working sample, 25% test split). Without SMOTE, every algorithm achieves relatively high accuracy Eq. (8): (0.769–0.787) but low recall Eq. (9): (0.330–0.442), indicating that unmodified classifiers default to predicting the majority (Normal/Low-Demand) class and largely fail to detect genuine high-demand sessions, a textbook symptom of training on imbalanced data [9]. Applying SMOTE consistently and substantially increases recall across all five algorithms (up to +0.366 for XGBoost), at the cost of precision and a modest drop in raw accuracy, which is the expected precision–recall trade-off of minority-class oversampling [10].

ISSN: 2715-6427

Table 1. Ablation results.

Model	Condition	Accuracy	Precision	Recall	F1-score	ROC-AUC
Decision Tree	No SMOTE	0.778	0.576	0.416	0.483	0.808
Decision Tree	SMOTE	0.724	0.470	0.813	0.596	0.806
K-Nearest Neighbors	No SMOTE	0.769	0.548	0.436	0.486	0.808
K-Nearest Neighbors	SMOTE	0.720	0.463	0.761	0.576	0.797
Logistic Regression	No SMOTE	0.787	0.645	0.330	0.436	0.820
Logistic Regression	SMOTE	0.757	0.510	0.713	0.595	0.822
Random Forest	No SMOTE	0.774	0.567	0.411	0.477	0.816
Random Forest	SMOTE	0.731	0.477	0.795	0.596	0.815
XGBoost	No SMOTE	0.777	0.570	0.442	0.498	0.818
XGBoost	**SMOTE**	0.731	0.478	**0.807**	**0.600**	0.818

Ranked by F1-score Eq. (10) **XGBoost + SMOTE** is the best-performing configuration (F1 = 0.600, ROC-AUC = 0.818), narrowly ahead of Random Forest + SMOTE (F1 = 0.596) and Decision Tree + SMOTE (F1 = 0.596). The metric comparison in Figure 6 visualizes this F1 advantage, while Figure 7 shows that ROC-AUC, a threshold-independent measure, remains almost unchanged by SMOTE for every algorithm (≤0.02 difference). Figure 8 further illustrates the operational trade-off behind this result: SMOTE reduces false negatives and increases recall, but at the cost of additional false positives. This pattern, and XGBoost's edge over the other four algorithms, is consistent with prior EV-specific studies that likewise found gradient-boosted trees to outperform simpler classifiers on charging-behavior prediction tasks [6], [7].

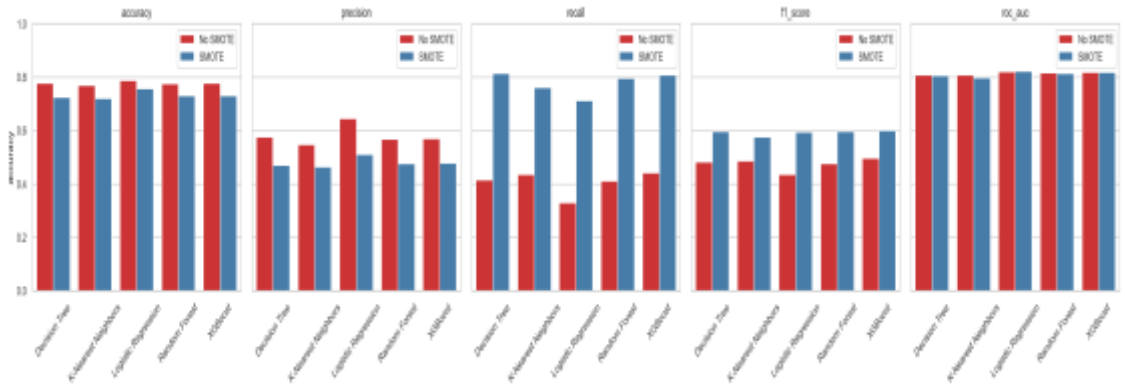

Figure 6. Ablation comparison.

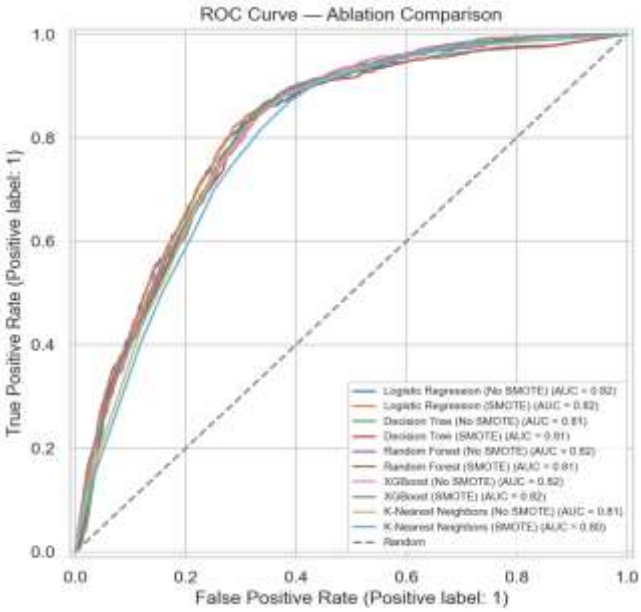

Figure 7. ROC curves for all ten model/condition combinations.

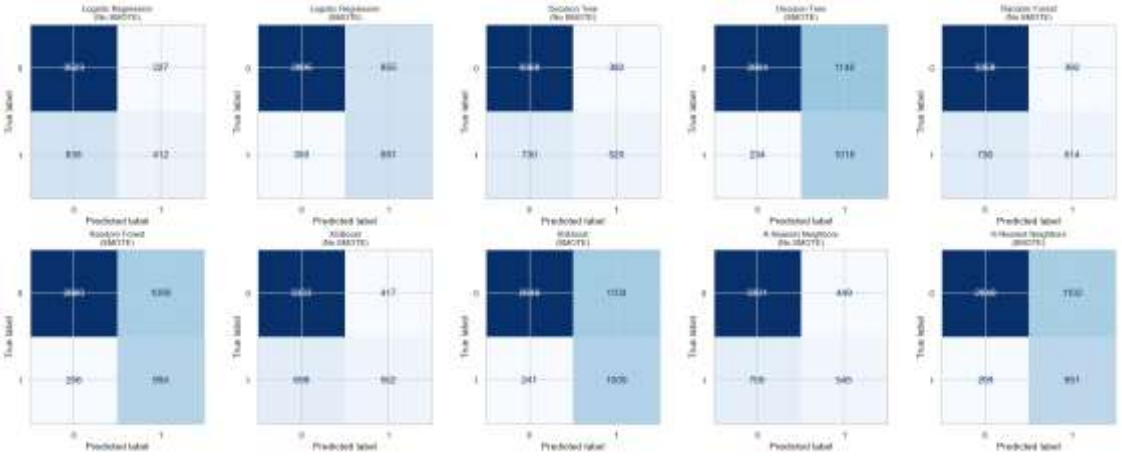

Figure 8. Confusion matrices for all models.

4.3. Explainable AI (XAI) Interpretation

The ablation results in Section 4.2 establish *which* configuration wins; the SHAP analysis below examines how that model distributes feature attribution. Figure 9 shows the SHAP summary plot Eq. (11) for the best model (XGBoost + SMOTE). chargingDuration is by far the dominant predictor in the trained model: high duration values (red) push predictions strongly toward the high-demand class, while low duration values (blue) push predictions toward normal/low demand, consistent with the moderate positive correlation observed in Section 3.2 ($r = 0.51$). connectionTime_decimal is the second most influential feature, while is_peak_hour and is_weekend contribute comparatively little, aligning with the near-zero raw correlation of dayIndicator-derived features with delivered energy. Figure 10 provides the corresponding tree-based feature-importance view for the same XGBoost + SMOTE model, and an independent permutation-importance check on the Logistic Regression model produced the same feature ranking (chargingDuration $\gg$ connectionTime_decimal > is_peak_hour > is_weekend), reinforcing the stability of this model-level interpretation across model families [12], [23].

ISSN: 2715-6427

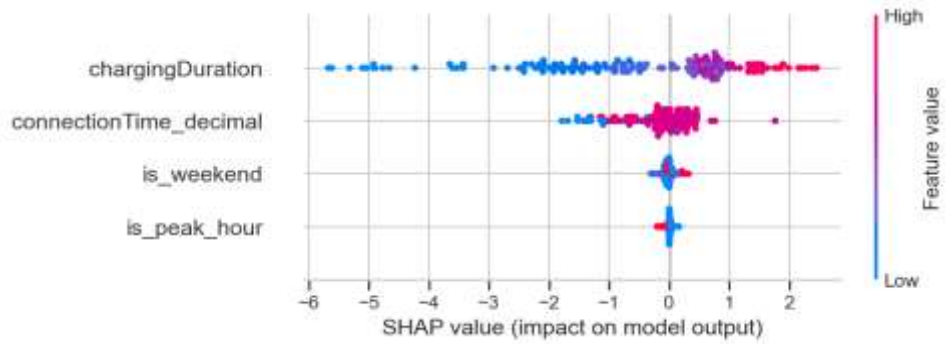

Figure 9. SHAP summary plot for the best-performing configuration.

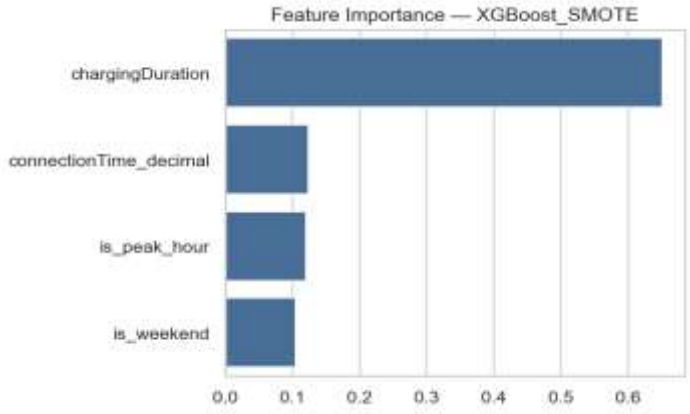

Figure 10. Tree-based feature importance.

4.4. Deployment: Interactive Web Application

Every number and figure discussed in Sections 4.1–4.3 was produced not by a one-off script, but by the same deployed application a grid planner would use. Figure 11 shows the Evaluation tab of the deployed Streamlit application after a live pipeline run, displaying the best-model summary banner, the ablation comparison chart, and the full confusion-matrix grid generated in real time from user-selected sample size and test-split settings. All ten CSV/PNG artifacts referenced in Sections 4.1–4.3 are produced identically whether the pipeline is triggered through this interface or run as a standalone script, satisfying the reproducibility goal set out in Section 2.5.

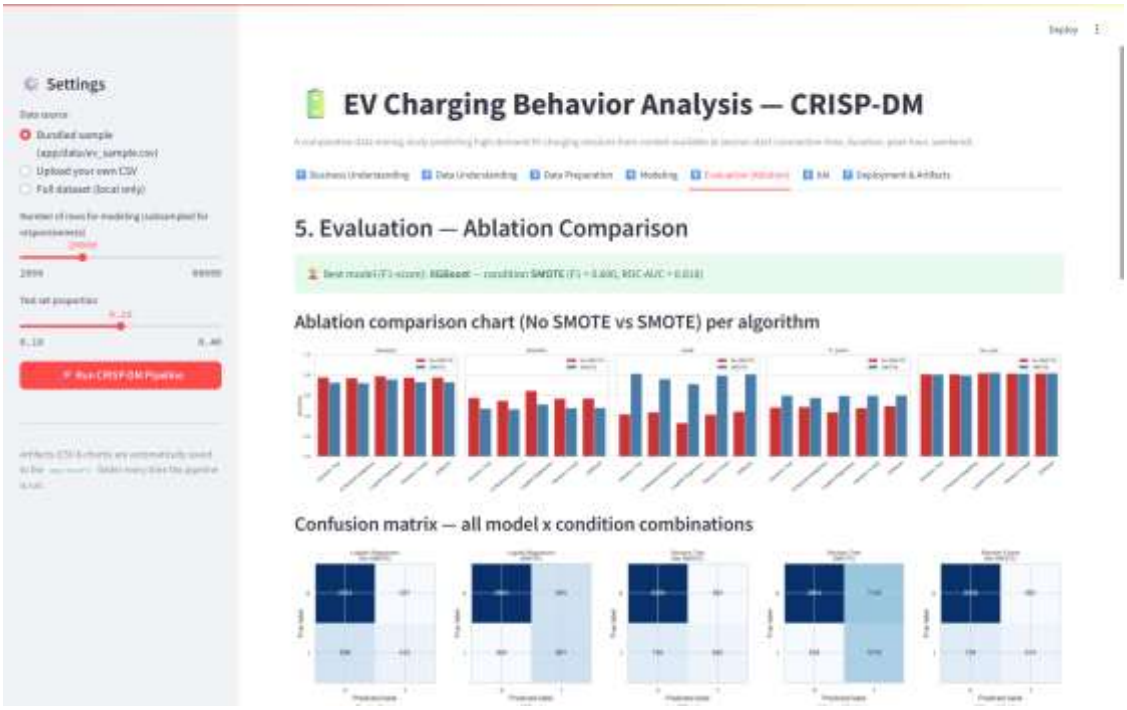

Figure 11. Evaluation (Ablation) tab of the deployed Streamlit application.

5. CONCLUSION

This study produced a CRISP-DM–structured, leakage-aware pipeline for early identification of high-demand EV charging sessions from context-only information available at connection time. The main empirical finding is that applying SMOTE substantially improves minority-class detection, with XGBoost + SMOTE giving the best overall F1-score (0.600) while maintaining ROC-AUC = 0.818. SHAP-based attribution indicated that the trained model relies most strongly on planned charging duration and connection time, while the sequential day index contributes little through the derived weekend feature; this should be interpreted as model-level attribution rather than causal evidence. The resulting Streamlit application packages the analysis into an auditable artifact that archives the quantitative and visual outputs for reuse by non-programmer stakeholders such as grid planners. This work has two main limitations that motivate future research. First, the underlying dataset is synthetic; validating the pipeline on real-world session data, now increasingly available in high-resolution transaction form, is needed before operational deployment. Second, the classification framing (binary, quartile-based) discards information relative to a continuous energy-demand regression or multi-class demand-tier model, which could offer finer-grained operational guidance. Future work will extend the pipeline to real-world charging logs, add regression-based energy forecasting alongside the current classifier, and evaluate additional imbalance-correction techniques (e.g., ADASYN, cost-sensitive learning) within the same ablation framework.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to the Information Systems Engineering Laboratory, Campus 4 PSDKU Sidoarjo, Politeknik Negeri Jember, for providing the resources that supported the development of this research.

REFERENCES

[1] S. Ghanbari Motlagh and L. Li, "A review on electric vehicle charging station planning: Infrastructure placement, sizing, upgrades, and uncertainties," *J. Energy Storage*, vol. 141, p. 119325, 2026, doi: 10.1016/j.est.2025.119325.

[2] S. Mohanty *et al.*, "Demand side management of electric vehicles in smart grids: A survey on strategies, challenges, modeling, and optimization," *Energy Reports*, vol. 8, pp. 12466–12490, 2022, doi: 10.1016/j.egyr.2022.09.023.

[3] S. Fatima, V. Püvi, M. Lehtonen, and M. Pourakbari-Kasmaei, "A review of electric vehicle hosting

capacity quantification and improvement techniques for distribution networks," *IET Gener. Transm. Distrib.*, vol. 18, no. 6, pp. 1095–1113, 2024, doi: 10.1049/gtd2.13010.

[4] B. Palaniyappan and T. Vinopraba, "Dynamic pricing for load shifting: Reducing electric vehicle charging impacts on the grid through machine learning-based demand response," *Sustain. Cities Soc.*, vol. 103, p. 105256, 2024, doi: 10.1016/j.scs.2024.105256.

[5] G. Vishnu, D. Kaliyaperumal, P. B. Pati, A. Karthick, N. Subbanna, and A. Ghosh, "Short-term forecasting of electric vehicle load using time series, machine learning, and deep learning techniques," *World Electr. Veh. J.*, vol. 14, no. 9, p. 266, 2023, doi: 10.3390/wevj14090266.

[6] I. Ullah, K. Liu, T. Yamamoto, M. Zahid, and A. Jamal, "Modeling of machine learning with SHAP approach for electric vehicle charging station choice behavior prediction," *Travel Behav. Soc.*, vol. 31, pp. 78–92, 2023, doi: 10.1016/j.tbs.2022.11.006.

[7] T. Zhang, Q. Peng, and S. Zeng, "Predicting EV charging demand in renewable-energy-powered grids using explainable machine learning," *Sustainability*, vol. 17, no. 9, p. 4158, 2025, doi: 10.3390/su17094158.

[8] R. Alsaigh, R. Mehmood, and I. Katib, "AI explainability and governance in smart energy systems: A review," *Front. Energy Res.*, vol. 11, p. 1071291, 2023, doi: 10.3389/fenrg.2023.1071291.

[9] W. Chen, K. Yang, Z. Yu, Y. Shi, and C. L. P. Chen, "A survey on imbalanced learning: Latest research, applications and future directions," *Artif. Intell. Rev.*, vol. 57, p. 137, 2024, doi: 10.1007/s10462-024-10759-6.

[10] A. Arafa, N. El-Fishawy, M. Badawy, and M. Radad, "RN-SMOTE: Reduced noise SMOTE based on DBSCAN for enhancing imbalanced data classification," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5059–5074, 2022, doi: 10.1016/j.jksuci.2022.06.005.

[11] A. Saranya and R. Subhashini, "A systematic review of explainable artificial intelligence models and applications: Recent developments and future trends," *Decis. Anal. J.*, vol. 7, p. 100230, 2023, doi: 10.1016/j.dajour.2023.100230.

[12] A. V Ponce-Bobadilla, V. Schmitt, C. S. Maier, S. Mensing, and S. Stodtmann, "Practical guide to SHAP analysis: Explaining supervised machine learning model predictions in drug development," *Clin. Transl. Sci.*, vol. 17, no. 11, p. e70056, 2024, doi: 10.1111/cts.70056.

[13] V. Plotnikova, M. Dumas, and F. P. Milani, "Applying the CRISP-DM data mining process in the financial services industry: Elicitation of adaptation requirements," *Data Knowl. Eng.*, vol. 139, p. 102013, 2022, doi: 10.1016/j.datak.2022.102013.

[14] G. Douzas, "imbalanced-learn-extra: A Python package for novel oversampling algorithms," *J. Open Res. Softw.*, 2026, doi: 10.5334/jors.459.

[15] D. Kreuzberger, N. Kühl, and S. Hirschl, "Machine learning operations (MLOps): Overview, definition, and architecture," *IEEE Access*, vol. 11, pp. 31866–31879, 2023, doi: 10.1109/access.2023.3262138.

[16] M. Zarour, H. Alzabut, and K. T. Al-Sarayreh, "MLOps best practices, challenges and maturity models: A systematic literature review," *Inf. Softw. Technol.*, vol. 183, p. 107733, 2025, doi: 10.1016/j.infsof.2025.107733.

[17] L. Pilgram, H. Ko, A. Tung, and K. El Emam, "Protecting patient privacy in tabular synthetic health data: A regulatory perspective," *npj Digit. Med.*, vol. 8, p. 732, 2025, doi: 10.1038/s41746-025-02112-0.

[18] Y. Zhang *et al.*, "A high-resolution electric vehicle charging transaction dataset with multidimensional features in China," *Sci. Data*, vol. 12, p. 643, 2025, doi: 10.1038/s41597-025-04982-1.

[19] S. Kapoor and A. Narayanan, "Leakage and the reproducibility crisis in machine-learning-based science," *Patterns*, vol. 4, no. 9, p. 100804, 2023, doi: 10.1016/j.patter.2023.100804.

[20] S. Somvanshi, S. Das, S. Javed, G. Antariksa, and A. Hossain, "A survey on tabular data: From tree-based methods to tabular deep learning," *ACM Comput. Surv.*, 2026, doi: 10.1145/3807777.

[21] F. E. Arévalo-Cordovilla and M. Peña, "Comparative analysis of machine learning models for predicting student success in online programming courses: A study based on LMS data and external factors," *Mathematics*, vol. 12, no. 20, p. 3272, 2024, doi: 10.3390/math12203272.

[22] V. G. Costa and C. E. Pedreira, "Recent advances in decision trees: An updated survey," *Artif. Intell. Rev.*, vol. 56, pp. 4765–4800, 2023, doi: 10.1007/s10462-022-10275-5.

[23] M. Aria, C. Cuccurullo, and A. Gnasso, "A comparison among interpretative proposals for random forests," *Mach. Learn. with Appl.*, vol. 6, p. 100094, 2021, doi: 10.1016/j.mlwa.2021.100094.

[24] G. Velarde *et al.*, "Tree boosting methods for balanced and imbalanced classification and their robustness over time in risk assessment," *Mach. Learn. with Appl.*, 2025, doi: 10.1016/j.iswa.2024.200354.

[25] A. Ali, M. Hamraz, N. Gul, D. M. Khan, S. Aldahmani, and Z. Khan, "A k nearest neighbour ensemble via extended neighbourhood rule and feature subsets," *Pattern Recognit.*, vol. 142, p. 109641, 2023, doi: 10.1016/j.patcog.2023.109641.

[26] K. Hamad, L. Obaid, S. Haridy, W. Zeiada, and G. G. Al-Khateeb, "Factorial design-machine learning approach for predicting incident durations," *Comput. Civ. Infrastruct. Eng.*, vol. 38, no. 5, pp. 660–680, 2023, doi: 10.1111/mice.12883.

[27] A. F. A. Sayed, M. A. Arafa, N. A. El-Nimr, K. A. S. Banawan, and M. S. Abdou, "Comparison of machine learning classification and regression models for prediction of academic performance among postgraduate public health students," *Sci. Rep.*, vol. 15, p. 44056, 2025, doi: 10.1038/s41598-025-31023-z.

[28] T. W. Campbell, H. Roder, R. W. Georgantas III, and J. Roder, "Exact Shapley values for local and model-true explanations of decision tree ensembles," *Mach. Learn. with Appl.*, vol. 9, p. 100345, 2022, doi: 10.1016/j.mlwa.2022.100345.

[29] A. B. Setiawan, H. Y. Riskiawan, H. A. Putranto, T. Rizaldi, and R. A. Atmoko, "An end-to-end machine learning pipeline for online purchase intention prediction using Random Forest and MLOps practices," *Angkasa J. Ilm. Bid. Teknol.*, vol. 18, no. 1, pp. 10–20, 2026, doi: 10.28989/angkasa.v18i1.3841.

[30] S. P. Revathy, M. Sindhuja, and R. Jayashree, "Streamlit-based web application for Parkinson's detection using machine learning," *J. Artif. Intell. Capsul. Networks*, vol. 6, no. 4, pp. 415–428, 2024, doi: 10.36548/jaicn.2024.4.006.

[31] A. R. Singh, R. S. Kumar, M. Bajaj, C. B. Khadse, and I. Zaitsev, "Machine learning-based energy management and power forecasting in grid-connected microgrids with multiple distributed energy sources," *Sci. Rep.*, vol. 14, 2024, doi: 10.1038/s41598-024-70336-3.

[32] A. Lavanya, S. Ali, and S. D. Vidya Sagar, "Assessing the performance of Python data visualization libraries: A review," *Int. J. Comput. Eng. Res. Trends*, vol. 10, no. 1, pp. 29–39, 2023, doi: 10.22362/ijcert/2023/v10/i01/v10i0104.

[33] F. Nahrstedt, M. Karmouche, K. Bargieł, P. Banijamali, A. N. Pradeep Kumar, and I. Malavolta, "An empirical study on the energy usage and performance of Pandas and Polars data analysis Python libraries," 2024, *Salerno, Italy*. doi: 10.1145/3661167.3661203.

[34] A. B. Setiawan, H. Y. Riskiawan, T. Rizaldi, H. A. Putranto, R. A. Atmoko, A. B. F. Mansur, M. Alharthi, and A. H. Basori, "From chaos to Gleichgewicht: An application-specific BiLSTM framework with SMOTE, SHAP, and TF-IDF-based safety support for proxy risk screening on noisy social media," *Computation*, vol. 14, no. 8, art. 167, 2026, https://doi.org/10.3390/computation14080167.

[35] D. P. S. Setyohadi, H. Y. Riskiawan, A. S. Arifianto, I. G. Wiryawan, and A. B. Setiawan, "Explainable clinical-operational intelligence for hospital length of stay prediction using integrated multi-source admission data with time-based evaluation," *J. Vocational Inform. Comput. Educ.*, vol. 4, no. 2, 2026, https://doi.org/10.66053/voice.v4i2.507.