PreXP: Enhancing Trust in Data Preprocessing Through Explainability

Sandra Samuel[a] and Nada Sharaf[b]
Department of Informatics and Computer Science, German International University, Cairo, Egypt

Keywords: Automated Preprocessing, Explainable AI, User Engagement, Machine Learning, Large Language Model.

Abstract: Data preprocessing is a crucial yet often opaque stage in machine learning workflows. Manual preprocessing is time consuming and inconsistent, while automated pipelines efficiently transform data but lack explainability, making it difficult to track modifications and understand preprocessing decisions. This lack of transparency can lead to uncertainty and reduced confidence in data preparation. PreXP (Preprocessing with Explainability) addresses this gap by enhancing transparency in preprocessing workflows. Rather than modifying data, PreXP provides interpretability by documenting and clarifying preprocessing steps, ensuring that users remain informed about how their data has been prepared. Initial evaluations suggest that increasing visibility into preprocessing decisions improves trust and interpretability, reinforcing the need for explainability in data driven systems and supporting the development of more accountable machine learning workflows.

1 INTRODUCTION

Data preprocessing is a critical stage in machine learning workflows, transforming raw data into a structured format suitable for modeling. It directly influences data quality, model performance, and the reliability of insights derived from machine learning systems. Despite its significance, preprocessing is often treated as an opaque process, where transformations occur without explicit documentation or transparency. While automation has improved efficiency, it frequently lacks explainability, leaving users unaware of the modifications applied to their data.

Existing preprocessing approaches pose significant challenges. Manual preprocessing, although providing control and flexibility, is time intensive and prone to inconsistencies. In contrast, automated tools efficiently transform data but rarely offer insights into their decisions. As a result, users may receive preprocessed data without clarity on what changes were made, making it difficult to verify transformations and assess their impact. The absence of explainability in preprocessing creates a barrier to trust and interpretability in data driven workflows.

Explainability in preprocessing is essential for ensuring transparency in machine learning pipelines. While extensive research has focused on explainability in model decision making, preprocessing remains an overlooked aspect despite its direct impact on downstream results. A clear understanding of data transformations allows users to verify integrity, assess biases, and make informed decisions before deploying models.

To address this gap, PreXP (Preprocessing with Explainability) is introduced as a preprocessing tool that integrates automation with transparency. It executes key preprocessing tasks including handling missing data, encoding categorical variables, and scaling numerical features while simultaneously documenting and explaining each step. Unlike conventional tools that apply transformations without visibility, this framework ensures that preprocessing decisions remain interpretable, traceable, and accessible to users.

This paper evaluates the effectiveness of the proposed solution in enhancing transparency in data preparation. It explores the core functionalities of the framework, discusses its role in improving trust in machine learning workflows, and highlights the importance of ensuring that data transformations are both effective and clearly understood.

2 RELATED WORK

The automation of data preprocessing is a key component in modern machine learning pipelines, directly influencing model performance and interpretability.

[a] https://orcid.org/0009-0004-9882-8726
[b] https://orcid.org/0000-0002-0681-6743

Samuel, S., Sharaf and N.
PreXP: Enhancing Trust in Data Preprocessing Through Explainability.
DOI: 10.5220/0013519400003967
In *Proceedings of the 14th International Conference on Data Science, Technology and Applications (DATA 2025)*, pages 407-414
ISBN: 978-989-758-758-0; ISSN: 2184-285X
Copyright © 2025 by Paper published under CC license (CC BY-NC-ND 4.0)

Despite its importance, preprocessing remains largely opaque, with automated tools transforming data without transparency. While automation and generative AI techniques have improved preprocessing efficiency, challenges related to interpretability, scalability, and generalizability persist. This section reviews key developments in automated preprocessing frameworks, profiling integration, and the lack of explainability, positioning PreXP as a transparent alternative. There have been previous work for the generalization of visualizations and the automated generation of them as well (AbouWard et al., 2024; Roshdy et al., 2018).

2.1 Automation Frameworks for Preprocessing

Several frameworks have been proposed to reduce manual preprocessing and enhance consistency. Giovanelli et al. (Giovanelli et al., 2021a) showed that optimized pipelines improve predictive performance, yet transformation sequences remain difficult to interpret. AutoGluon Tabular (Erickson et al., 2020) automates preprocessing for structured data but lacks visibility into applied transformations. Atlantic (Santos and Ferreira, 2023) adapts pipelines dynamically but does not explain decisions. Similarly, Preprocessy (Kazi et al., 2022) and the pipeline from (Chheda et al., 2021) prioritize flexibility and efficiency, while omitting transformation level interpretability.

2.2 Integrating Profiling with Preprocessing

Data profiling plays a critical role in assessing dataset quality before transformation, yet it is often treated as a standalone process. Most existing frameworks overlook profiling as an integrated component of preprocessing, requiring users to manually interpret statistics before applying transformations. Salhi et al. (Salhi et al., 2023) highlighted that this disconnect can reduce efficiency and lead to suboptimal outcomes. Zakrisson et al. (Zakrisson, 2023) introduced the Trinary Decision Tree, which enhances missing value handling by classifying missingness types; however, it does not extend explainability to other preprocessing stages. As a result, transparency remains limited when users cannot trace how profiling insights inform transformation choices.

2.3 Challenges in Explainability for Preprocessing

Despite growing interest in XAI, preprocessing explainability remains underexplored. Most frameworks apply transformations without rationale, leaving users uncertain about how missing values were handled, how encoding was performed, or why certain scaling methods were used. Salhi et al. (Salhi et al., 2023) and Zakrisson et al. (Zakrisson, 2023) acknowledge this gap, while Tae et al. (Tae et al., 2019) highlight the difficulty of handling structured and unstructured data without losing explainability. Black box preprocessing introduces trust concerns that remain unresolved.

2.4 Positioning PreXP as a Transparent Preprocessing Framework for Structured Data

PreXP bridges the gap between automation and transparency by ensuring each preprocessing step is both applied and explained. Unlike existing tools that operate as black boxes, PreXP logs and justifies every transformation, allowing users to query specific steps such as missing value handling, encoding, and scaling.

A key distinction lies in its integration of profiling within the preprocessing workflow. Instead of requiring manual interpretation of statistics, PreXP uses profiling insights to guide decisions dynamically, making the process more efficient and interpretable. By offering explainable, queryable transformations, PreXP shifts preprocessing toward greater accountability. Future work will focus on incorporating domain specific modules and expanding its explanation capabilities to enhance user trust and interaction.

3 METHODOLOGY

PreXP was developed to automate preprocessing for structured datasets while ensuring full transparency of each transformation. Unlike conventional preprocessing tools that apply transformations without user insight, PreXP provides a detailed explanation of every step, allowing users to verify and understand data modifications. This section outlines the archictecture of the tool, workflow, preprocessing techniques, and explainability mechanisms.

3.1 System Architecture and Design

PreXP follows a structured workflow that balances efficient preprocessing with transparency. The architecture, illustrated in Figure 1, outlines the core components of the tool and data flow.

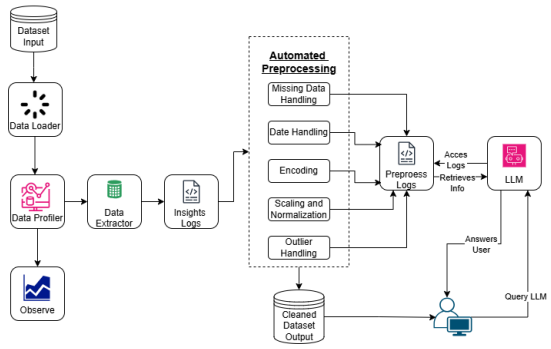

Figure 1: System workflow illustrating the preprocessing pipeline and explainability integration in PreXP.

The process begins with dataset upload, where structured CSV files are verified and profiled. Profiling generates statistical summaries, correlation matrices, and missingness categorizations (MCAR, MAR, MNAR), which are stored in the insights log to inform preprocessing steps.

Automated transformations including missing value handling, encoding, scaling, outlier treatment, and date feature extraction are then applied. Each step is recorded in the preprocessing log, detailing decisions such as imputation methods or encoding strategies.

To support explainability, PreXP maintains three distinct logs: (1) insights logs for dataset characteristics, (2) preprocessing logs for transformation details, and (3) query logs for user inquiries. These enable traceable, queryable explanations.

User queries are processed by a LLM, which retrieves relevant entries from the logs and generates natural language responses explaining the rationale behind each transformation.

After preprocessing, the transformed dataset is made available for download, enabling users to continue their machine learning workflows with full awareness of all modifications.

3.2 Tool Functions and Flow

This section outlines the structured workflow within PreXP, detailing its core functionalities and interaction flow.

1. Dataset Upload: Users upload a CSV file, which is verified before processing.
2. Profiling and Insights Generation: The dataset is analyzed, generating statistical summaries and missing data distributions.
3. Automated Preprocessing: Transformations such as missing data handling, encoding, scaling, and outlier management are applied.
4. Explainability and Query Mechanism: Preprocessing decisions are logged, enabling users to query and retrieve explanations.
5. Preprocessed Data Output: The final processed dataset is displayed for verification and download.

The following subsections expand on each step, illustrating the functionality and role in transparent preprocessing.

3.2.1 Dataset Uploading

PreXP opens with an introductory interface that outlines its core functionalities, including profiling, automated transformations, and explainability.

The first step involves uploading a structured CSV file. The system automatically verifies the format of the file, checks for missing headers, and ensures structural consistency. Users may optionally specify a target column to guide encoding and missing data handling; if not provided, preprocessing proceeds in a generic manner suitable for general purpose workflows.

3.2.2 Data Profiling

Data profiling systematically examines the dataset to build a comprehensive understanding of its structure and contents. It captures key statistics, including mean, median, minimum and maximum values, percentages of missing data, skewness, and the presence of outliers. These insights are stored in logs that highlight patterns in missingness and are later used to inform preprocessing decisions such as imputation, encoding, and scaling.

Upon completion, PreXP generates a detailed profiling report Figure 2 that includes numerical summaries, correlation heatmaps, and distribution plots. These visual components expose relationships between variables and help detect irregularities within the dataset, playing a crucial role in assessing data quality and preparing for transformation.

3.2.3 Automated Preprocessing

PreXP automates essential preprocessing steps to prepare raw data for machine learning workflows. The process includes handling missing data, encoding categorical variables, scaling numerical features, managing outliers, and extracting date related components.

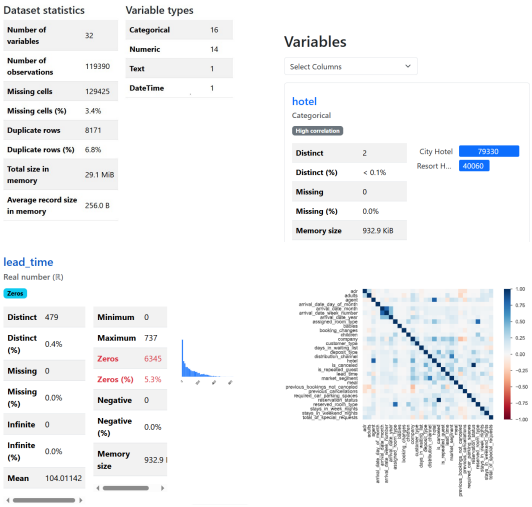

Figure 2: Compressed view of the profiling report showing dataset characteristics and relationships.

These transformations are dynamically tailored based on the dataset's characteristics, ensuring adaptability across diverse use cases.

Users are presented with a summary of each step applied and can review the final dataset upon completion, reinforcing transparency throughout the preprocessing pipeline. A detailed explanation of the underlying techniques follows in subsequent sections.

3.2.4 Explainability Visualizations

PreXP enhances transparency by offering detailed visualizations that summarize key preprocessing steps, allowing users to understand how their data has been transformed. These visual insights facilitate interpretability and ensure that preprocessing decisions remain traceable. The figures presented in this section illustrate a subset of the available explainability features, demonstrating key transformations applied to the dataset.

Figure 3 provides an analysis of missing data, distinguishing between different patterns observed within the dataset. It highlights the proportion of values categorized as Missing Completely at Random (MCAR) and Missing Not at Random (MNAR), aiding users in assessing potential biases introduced by missing values.

Figure 4 presents an overview of the encoding strategies applied to categorical variables. It illustrates the distribution of encoding methods such as one hot encoding, frequency encoding, and target based encoding, ensuring users can assess how categorical data has been transformed.

To complement the encoding overview, Figure 5 details encoding transformations at a column level,

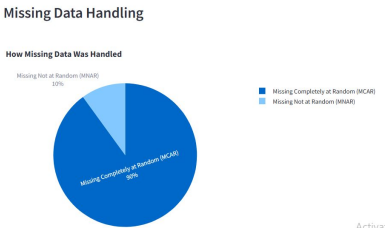

Figure 3: Analysis of missing data, displaying proportions of MCAR and MNAR values.

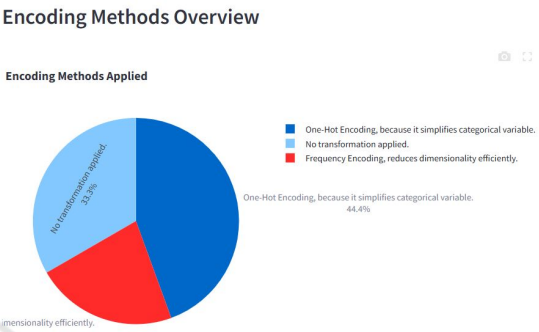

Figure 4: Overview of encoding techniques applied to categorical variables.

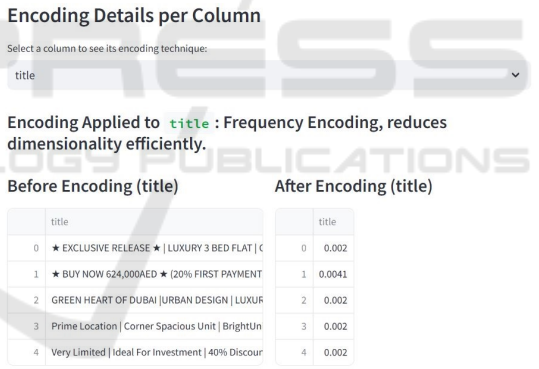

Figure 5: Column specific encoding transformations applied during preprocessing.

providing a breakdown of which categorical variables were assigned specific encoding techniques. This enables users to verify consistency in the encoding strategy across different features.

Beyond individual transformations, Figure 6 highlights the structured logging mechanism in PreXP, which systematically records every preprocessing step taken. This ensures that all modifications, such as missing data handling, encoding, and scaling, are documented for future reference.

Lastly, PreXP provides an interactive query interface Figure 7, allowing users to retrieve justifications for preprocessing actions. Powered by a LLM, this feature enables users to ask questions regarding spe-

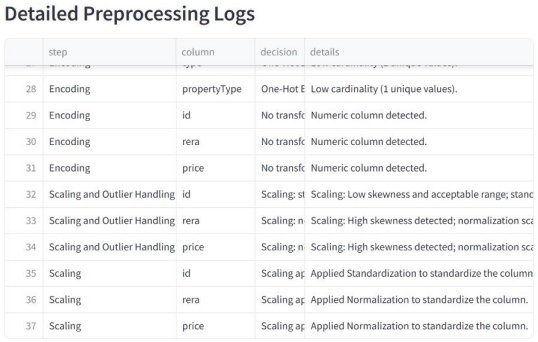

Figure 6: Preprocessing logs capturing applied transformations for transparency.

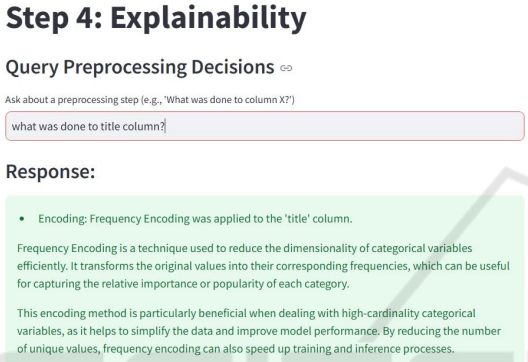

Figure 7: Query interface.

cific transformations and receive detailed responses, reinforcing transparency and trust in the preprocessing workflow.

4 BACKEND PROCESSING

This section explains the core mechanisms behind the automated preprocessing of PreXP and its explainability features. It details how transformations are applied based on structured rules and how the system enables explainability through a LLM.

4.1 Preprocessing Techniques

PreXP employs a structured and rule based approach to preprocessing, ensuring transformations are both automated and explainable. This section outlines the key operations applied to structured datasets.

Missing Data Handling

PreXP classifies missing values using the MCAR, MAR, and MNAR framework and applies appropriate strategies:

- **Deletion:** Rows or columns are removed if missing values exceed a defined threshold.
- **Imputation:** Numerical values are filled using mean, median, or regression based techniques:

$$x_{i,j} = f(x_{-i,j}) + \varepsilon \tag{1}$$

where $f(x_{-i,j})$ is a predictive function using available data.

- **Categorical Handling:** Missing categorical values are imputed with the mode or labeled as 'Missing'.

Encoding Categorical Data

Encoding is selected based on variable cardinality:

- **One Hot Encoding:** Generates binary indicators for low cardinality features:

$$z_{i,j} = \begin{cases} 1, & \text{if } x_j = c_i \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

- **Frequency Encoding:** Replaces categories with relative frequency:

$$x'_i = \frac{\text{Count of } x_i}{\text{Total Count}} \tag{3}$$

- **Target Encoding:** Maps categories to their average target value:

$$x'_i = \frac{\sum_{y \in C_i} y}{|C_i|} \tag{4}$$

Scaling and Normalization

To ensure numerical comparability, PreXP applies:

- **Min-Max Scaling:**

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{5}$$

- **Robust Scaling:** More resilient to outliers:

$$x' = \frac{x - \text{median}(x)}{\text{IQR}(x)} \tag{6}$$

Outlier Detection and Treatment

Using the IQR method, PreXP identifies and optionally removes or adjusts outliers:

$$\text{IQR} = Q_3 - Q_1 \tag{7}$$

$$\begin{aligned} \text{Lower Bound} &= Q_1 - 1.5 \times \text{IQR} \\ \text{Upper Bound} &= Q_3 + 1.5 \times \text{IQR} \end{aligned} \tag{8}$$

Date Feature Extraction

Date fields are decomposed into machine readable components:

- Year, Month, Day
- Day of week
- Time elapsed from a reference date

Together, these preprocessing techniques enable consistent and interpretable transformation of structured datasets. The next section discusses how these transformations are made transparent through PreXP's explainability system.

4.2 LLM Powered Explainability

A key challenge in automated preprocessing is ensuring that users understand the transformations applied to their data. PreXP addresses this by integrating a LLM, powered by Cohere, which enables users to query preprocessing steps and receive clear, contextual explanations. Rather than operating as a black box, PreXP offers direct justifications for each transformation. Cohere was selected for its freely available research API, making it well suited for experimental use.

4.2.1 Structured Logging for Transparency

To support explainability, PreXP maintains structured logs that capture essential details at every step. These logs serve as the foundation for retrieving relevant information when users request insights into data transformations.

1. Insight logs store key dataset characteristics from profiling, including missing data patterns, feature distributions, and detected anomalies.
2. Preprocessing logs record specific transformations applied, such as how missing values were handled, which encoding techniques were used, and what scaling method was applied.
3. Query logs track user inquiries and the system responses, ensuring that all explanations are backed by actual preprocessing steps taken.

These logs allow PreXP to provide precise, traceable justifications for each preprocessing decision.

4.2.2 Explainability Mechanism

The explainability system of PreXP responds to user queries about data transformations through a structured three step process:

1. Analyzing the query to identify the relevant preprocessing step.
2. Retrieving corresponding actions from structured preprocessing logs.
3. Generating a natural language explanation using the Cohere LLM. Prompts are automatically created based on the logs and user intent, such as: *"Explain why Min Max scaling was applied to column 'Age'"* or *"State why column 'Country' was frequency encoded."*

This mechanism keeps users informed of data modifications, reinforcing transparency and trust. By exposing preprocessing decisions, PreXP offers an interpretable and verifiable preprocessing framework.

5 RESULTS AND EVALUATION

This section presents the evaluation of PreXP based on two studies: (1) A comparative study assessing its preprocessing performance against manual methods and (2) A usability study evaluating engagement, explainability, and user perception.

5.1 Comparative Study: Manual vs. PreXP

To assess the impact of PreXP on preprocessing efficiency and model performance, 10 participants manually preprocessed datasets before applying PreXP to the same data. The evaluation focused on preprocessing time, accuracy differences, and transparency in preprocessing decisions.

At the time of this study, no publicly available preprocessing frameworks offered built in explainability features similar to PreXP. Therefore, a manual baseline was used as the primary point of comparison to reflect current common practice in data preprocessing.

5.1.1 Preprocessing Time and Efficiency

PreXP demonstrated a significant reduction in preprocessing time, automating tasks that typically require substantial manual effort. Participants reported that manual preprocessing took an average of 33.43 minutes, whereas PreXP completed the same tasks in just 1.86 minutes, achieving a 94.44% time reduction. The automation of missing value handling, encoding, and scaling contributed to this efficiency.

Table 1: Preprocessing Time Comparison.

Metric	Value
Average Manual Preprocessing Time (mins)	33.43
Average PreXP Preprocessing Time (mins)	1.86
Time Reduction (%)	94.44

5.1.2 Model Performance Analysis

While PreXP does not yet include feature engineering, it maintained a reasonable level of accuracy across diverse datasets. In some cases, PreXP preprocessing improved accuracy, particularly in datasets requiring structured feature transformations. For example, in the *Predict Students Dropout* dataset, PreXP preprocessing increased accuracy from 0.75 to 0.79. Similarly, in *Stroke Prediction*, PreXP outperformed manual preprocessing in Logistic Regression (0.7671 vs. 0.7358) and Decision Trees (0.8309 vs. 0.6888).

The datasets evaluated in this experiment were independently chosen by the participants based on their familiarity and previous work. This approach ensured that participants were confident in their manual preprocessing choices, while also demonstrating the flexibility of PreXP across a wide range of real world datasets as shown in Table 2.

In some cases, manual preprocessing held a slight advantage due to adjustments specific to the dataset. For instance, in the *Real Estate UAE* dataset, manual steps led to marginally higher accuracy, highlighting the potential benefit of incorporating domain-specific feature engineering in future versions of PreXP.

Table 2: Model performance comparison: Manual preprocessing vs. PreXP.

Dataset	Model	Manual Accuracy	PreXP Accuracy
Mobile Price Classification	Random Forest	0.88	**0.89**
Plane Survival	Logistic Regression	**0.815**	0.803
Plane Survival	Random Forest	**0.8044**	0.786
Financial Loan Approval	Logistic Regression	**0.8113**	0.8018
Financial Loan Approval	XGBoost	**0.7830**	0.7477
Financial Loan Approval	Random Forest	**0.7925**	0.7748
Airline Delay Prediction	Logistic Regression	**0.6400**	0.5929
Airline Delay Prediction	XGBoost	0.6493	**0.7063**
Airline Delay Prediction	Random Forest	0.6589	**0.7021**
Predict Students Dropout	Decision Trees	0.75	**0.79**
Stroke Prediction	Logistic Regression	0.7358	**0.7671**
Stroke Prediction	Decision Tree	0.6888	**0.8309**
Stroke Prediction	Random Forest	**0.9412**	0.8499
Student Course Recommendation	Logistic Regression	0.7358	**0.7671**
Student Course Recommendation	Decision Tree	0.6888	**0.8309**
Student Course Recommendation	Random Forest	**0.9412**	0.8499
Real Estate UAE	Logistic Regression	**0.84**	0.82

5.1.3 Explainability and Preprocessing Differences

Explainability played a crucial role in participant assessments. The majority of participants (4.33/5 average rating) agreed that PreXP provided meaningful insights into preprocessing decisions. Every participant (100%) noticed differences between their manual preprocessing and the approach of PreXP, particularly in missing value handling and encoding strategies.

Several users noted that PreXP *"removed low variance columns"* and applied *"automated encoding decisions that differed from manual preprocessing"*. However, participants generally found the explainability features of PreXP useful in understanding transformations.

5.2 Usability Study: User Engagement and Explainability

Following the comparative study, a second experiment evaluated the usability, engagement, and explainability of PreXP. Twenty five participants from engineering and computer science backgrounds interacted with the tool using a standardized hotel booking dataset to ensure consistency across tests.

5.2.1 User Engagement and Perceived Usability

PreXP received positive feedback on usability: 84% of participants indicated they would use it frequently, and 76% strongly agreed it was easy to learn, reflecting its intuitive design. Confidence in using the tool was also high, with 60% feeling very confident and 40% expressing moderate confidence. Additionally, 64% strongly agreed that the tool's features were well integrated.

Most participants found PreXP easy to operate without external support, and 92% disagreed with the statement that it was unnecessarily complex. These ratings suggest that PreXP offers a smooth, user friendly experience requiring minimal onboarding.

5.2.2 Explainability and User Control

Participants gave PreXP high marks for explainability, with an average rating of 4.33 out of 5. A majority (64% strongly agreed, 28% agreed) felt in control of the preprocessing process. Furthermore, 82% found the explanation mechanism both engaging and effective.

Nearly all participants (98%) confirmed they understood what actions to take during tasks, supported by PreXP's query-based interface. Some participants recommended more detailed numerical justifications for steps such as encoding and scaling to further improve clarity.

Final Assessment: The usability study confirms the strong potential of PreXP for adoption due to its transparency, ease of use, and intuitive workflow. Future enhancements in feature engineering and deeper explainability may further elevate its value in data preprocessing pipelines.

6 CONCLUSIONS AND FUTURE WORK

PreXP has demonstrated strong performance in automating essential preprocessing steps.Its structured workflow minimizes manual effort, while integrated explainability features enhance user trust by clarifying preprocessing decisions.

PreXP currently lacks advanced feature engineering and domain specific customization. It also relies on external LLMs for explanations, which may vary by provider. The comparative study showed a 94.44% reduction in preprocessing time with accuracy comparable to manual methods, though domain specific cases showed slight benefits from manual tailoring. The usability study confirmed the clarity and trans-

parency of PreXP, with positive feedback on its interface and explanations.

Future enhancements will focus on advanced feature engineering (interaction detection, dimensionality reduction, feature selection) with user options. Generative AI will enhance imputation, data augmentation, and schema transformation. Explainability will improve with numerical justifications and visual aids. Further developments include refining preprocessing suggestions, cloud scalability, and benchmarking to establish PreXP as a robust, domain-aware tool. Participants proposed several enhancements for future iterations including: Advanced Feature Engineering for automated feature selection as well as expanding explainability to have precise numerical justifications for preprocessing decisions.

ACKNOWLEDGMENT

We acknowledge the use of AI tools to generate and enhance parts of the paper. The content was revised.

REFERENCES

AbouWard, F., Salem, A., and Sharaf, N. (2024). Autovi: Empowering effective tracing and visualizations with ai. In *2024 28th International Conference Information Visualisation (IV)*, pages 294–297. IEEE.

Balducci, F., Impedovo, D., and Pirlo, G. (2018). Machine learning applications on agricultural datasets for smart farm enhancement. *Machines*, 6(3):38.

Brown, P. A. and Anderson, R. A. (2023). A methodology for preprocessing structured big data in the behavioral sciences. *Behavior Research Methods*, 55(4):1818–1838.

Chheda, V., Kapadia, S., Lakhani, B., and Kanani, P. (2021). Automated data driven preprocessing and training of classification models. In *2021 4th International Conference on Computing and Communications Technologies (ICCCT)*, pages 27–32. IEEE.

Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., and Smola, A. (2020). Autogluon-tabular: Robust and accurate automl for structured data. *arXiv preprint arXiv:2003.06505*.

García, S., Ramírez-Gallego, S., Luengo, J., Benítez, J. M., and Herrera, F. (2016). Big data preprocessing: methods and prospects. *Big data analytics*, 1:1–22.

Giovanelli, J., Bilalli, B., and Abelló Gamazo, A. (2021a). Effective data pre-processing for automl. In *Proceedings of the 23rd International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data (DOLAP): co-located with the 24th International Conference on Extending Database Technology and the 24th International Conference on Database Theory (EDBT/ICDT 2021): Nicosia, Cyprus, March 23, 2021*, pages 1–10. CEUR-WS. org.

Giovanelli, J., Bilalli, B., and Gamazo, A. A. (2021b). Effective data pre-processing for automl. In *DOLAP'21, 23rd International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data*, pages 1–10, Nicosia, Cyprus. CEUR-WS.org.

Goyal, M. and Mahmoud, Q. H. (2024). A systematic review of synthetic data generation techniques using generative ai. *Electronics*, 13(17):3509.

Kaswan, K. S., Dhatterwal, J. S., Malik, K., and Baliyan, A. (2023). Generative ai: A review on models and applications. In *2023 International Conference on Communication, Security and Artificial Intelligence (ICCSAI)*, pages 699–704. IEEE.

Kazi, S., Vakharia, P., Shah, P., Gupta, R., Tailor, Y., Mantry, P., and Rathod, J. (2022). Preprocessy: a customisable data preprocessing framework with high-level apis. In *2022 7th international conference on data science and machine learning applications (CDMA)*, pages 206–211. IEEE.

Mishra, P., Biancolillo, A., Roger, J. M., Marini, F., and Rutledge, D. N. (2020). New data preprocessing trends based on ensemble of multiple preprocessing techniques. *TrAC Trends in Analytical Chemistry*, 132:116045.

Moore, R. and Lopes, J. (1999). Paper templates. In *TEMPLATE'06, 1st International Conference on Template Production*. SCITEPRESS.

Roshdy, A., Sharaf, N., Saad, M., and Abdennadher, S. (2018). Generic data visualization platform. In *2018 22nd International Conference Information Visualisation (IV)*, pages 56–57. IEEE.

Salhi, A., Henslee, A. C., Ross, J., Jabour, J., and Dettwiller, I. (2023). Data preprocessing using automl: A survey. In *2023 Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE)*, pages 1619–1623. IEEE.

Santos, L. and Ferreira, L. (2023). Atlantic—automated data preprocessing framework for supervised machine learning. *Software Impacts*, 17:100532.

Smith, J. (1998). *The Book*. The publishing company, London, 2nd edition.

Tae, K. H., Roh, Y., Oh, Y. H., Kim, H., and Whang, S. E. (2019). Data cleaning for accurate, fair, and robust models: A big data-ai integration approach. In *Proceedings of the 3rd international workshop on data management for end-to-end machine learning*, pages 1–4.

Varma, D., Nehansh, A., and Swathy, P. (2023). Data preprocessing toolkit: An approach to automate data preprocessing. *Interantional J. Sci. Res. Eng. Manag*, 7(03):15.

Westphal, P., Bühmann, L., Bin, S., Jabeen, H., and Lehmann, J. (2019). Sml-bench–a benchmarking framework for structured machine learning. *Semantic Web*, 10(2):231–245.

Zakrisson, H. (2023). Trinary decision trees for missing value handling. *arXiv preprint arXiv:2309.03561*.

Zhang, H., Dong, Y., Xiao, C., and Oyamada, M. (2023). Jellyfish: A large language model for data preprocessing. arxiv abs/2312.01678 (2023).