Robotic Process Automation (RPA) Scripting Model Using Machine Learning (ML) for Enterprise Data Validation and Integration

Luis Ángel Bendezú Jiménez, Jorge Luis Juan de Dios Apaza, Ruben Oscar Cerda García
Peruvian University of Applied Sciences, Lima, Perú

***Abstract*—This research presents an automated data processing model based on RPA Scripting, designed to enhance efficiency in extracting, validating, and integrating information from various web platforms. The automated workflow begins with the use of a tool that simulates human interaction on web platforms to obtain data automatically and reliably. The data is then organized and cleaned using processing techniques that prepare it for analysis. As a key component of the model, Machine Learning algorithms have been incorporated to detect errors, identify unusual patterns, and classify records, thereby improving data quality before storage. Finally, the processed data is loaded into a database and visualized through a dynamic dashboard that supports decision-making via reports and indicators. In conclusion, integrating Machine Learning algorithms within an RPA Scripting model not only optimizes the execution of automated tasks but also equips the model with intelligence to anticipate errors and adapt to changes in the data. This enables the development of a more robust, reliable, and adaptive automated process, aligned with current requirements for real-time analysis and decision-making.**

***Keywords*—RPA; scripting; data automation; Machine Learning; data validation; data integration; intelligent workflows**

I. Introduction

The volume of data generated by organizations has grown exponentially to the point that retrieving it from various web sources for the purpose of loading, integrating, and analyzing it requires the execution of manual processes that are both daily and repetitive. These processes represent a significant burden in terms of time and resources and are prone to human error, which can affect the efficiency and quality of the work. As a result, they also limit the agility and analytical capabilities of organizations operating in dynamic and competitive environments [1]. Moreover, these process flows do not end with data collection or consolidation; on the contrary, they extend into later stages such as interactive visualization, the generation of strategic reports, and advanced analysis within business intelligence platforms [2]. Therefore, any error or inefficiency in the initial stages of data extraction and processing can be amplified throughout the entire process flow, directly affecting the quality of data-driven strategic decisions [3].

This is interesting because it raises the opportunity to integrate traditionally separate technologies, such as robotic process automation (RPA) through Scripting and the application of Machine Learning (ML) algorithms to build process flows that not only execute daily and repetitive tasks, but also learn to make automatic and intelligent decisions. This convergence between RPA and ML opens up new possibilities to create more resilient, autonomous, and predictive solutions, overcoming the limitations of conventional automation tools. It also explores an emerging field that has enormous potential for application in multiple contexts.

Traditional automations fail due to their reliance on static rules and rigid structures, which do not allow them to adapt to dynamic changes in data or workflows. In [4], the authors point out how traditional RPA solutions present critical limitations when data sources change their structure, contain unexpected errors, or incomplete information. This is due to the fact that they are based only on rigid and predefined instructions, with no learning or adaptive capabilities. As a result, they cannot anticipate failures, detect anomalies, or validate data quality autonomously, which generates constant interruptions and requires manual intervention. These constraints reduce their scalability and reliability, especially in dynamic environments with high data variability. This problem highlights the need to adopt a more intelligent and robust approach, combining RPA with Machine Learning techniques to provide automated processes with predictive, adaptive, and decision-making capabilities based on historical patterns.

According to [5], they state that this problem has not been solved before because traditional automation solutions, such as conventional RPA tools, are designed to work under predefined rules and rigid structures. These solutions do not integrate advanced ML capabilities and adaptive learning, which makes them unable to handle changes in business data required for further analysis. In addition, commercial tools have a high cost and are oriented to users with general needs, which limits their customization for specific contexts and cases.

The key components of this approach focus on the integration of specialized Python libraries that allow the entire data integration flow to be intelligently automated. First, Playwright is used, which in this case plays a fundamental role within RPA scripting by facilitating dynamic interaction with web platforms, executing tasks such as browsing, authentication, and file download in a robust way. Once the data is obtained, Pandas comes into play, the library used for the cleaning, transformation, and validation of the extracted information, allowing for to structure the data in an optimal way for its analysis and subsequent integration. In addition, Oracledb is used to connect to the Oracle database and to load and insert processed data into multiple tables of the management system, ensuring persistence and consistency of the information. In addition, advanced ML techniques are incorporated to strengthen the quality and reliability of the process: Isolation Forest for the unsupervised detection of anomalies and subtle errors in rows; Autoencoders to identify deviations and unusual changes in the daily reports through deep learning, facilitating the generation of proactive alerts before insertion; and Random Forest for the classification of records into valid or invalid when historical tags are available, optimizing data cleansing. As a final stage of the flow, the validated and persisted data are displayed in enterprise visualization tools, which allow building interactive presentations, monitoring key indicators in real-time, and facilitating strategic decision-making based on reliable and updated information. This combination of tools allows end-to-end automation of a data integration flow, making it scalable, adaptable, and compatible with Machine Learning models to enrich operational intelligence in an automated and continuous way.

This study is structured as follows: Section II reviews related literature. Section III defines core concepts. Section IV details the proposed model architecture. Section V presents the experimental results. Section VI discusses limitations and future work. Finally, Section VII provides the concluding remarks.

II. Related Work

This section reviews previous research and technological developments related to RPA and ML integration techniques. The objective is to establish a theoretical basis for understanding existing solutions, identify current challenges, and highlight the gap that this study seeks to address. Through the analysis of studies in various industries, the convergence between automation and intelligent decision-making is discussed, highlighting the importance of building adaptive and efficient systems in data-driven environments.

In [6], the authors describe how RPA technology has revolutionized services globally, making it possible to replace the human labor force with software robots that incorporate intelligent elements, such as ML, to achieve more advanced and intelligent automation. In an industrial case in Finland, RPA is combined with ML to automate invoice processing in a shared services center, reducing errors, freeing up employees for higher-value tasks, and improving the efficiency and accuracy of the process. The integration of these technologies allows not only the automation of repetitive tasks but also the ability to learn and adapt to changes in data and processes, facilitating more intelligent and flexible automation. The study highlights that this combination can offer a good return on investment. However, this study's weakness is its focus on a single, internal business task (invoicing). It does not address the challenge of validating data extracted from multiple external web platforms, which is the gap this model addresses.

In [7], the authors explain how an RPA has been applied in the financial arena, specifically in predicting stock prices and automating the stock picking process. The research highlights a significant increase in demand for RPA-related jobs, reflecting a growth in the need to automate repetitive tasks in this sector. Price prediction models have been developed using Machine Learning techniques, such as stacked Long Short-Term Memory (LSTM). In addition, RPA's Automation Anywhere tool has been used to automate stock selection based on technical criteria, which helps users generate analysis reports in less time. While this demonstrates a powerful use of ML for prediction, its assumption is that the data is already clean. This approach is insufficient for our research, as this model focuses on validating the quality of internal enterprise data before it can be used for analysis.

In [8], the authors detail how intelligent robotic process automation (IRPA) combines artificial intelligence (AI) and RPA to automate complex unstructured tasks and adapt to changing scenarios. The research developed a process analysis framework for IRPA adoption, identifying key factors through a literature review and a survey of subject matter experts. The framework provides practical guidance for improving supply chain operations in areas such as logistics, purchasing, and inventory management. The strength of this framework is in optimizing unstructured processes. This differs from this work, which addresses the specific challenge of validating structured and semi-structured data extracted from web reports, a gap not covered by this supply chain model.

In [9], the authors state that sustainability in RPA refers to making RPA solutions robust and adaptable to change, enabling their long-term use and preventing them from ceasing to function when process changes occur. The research proposes a universal model for sustainable RPA implementation, consisting of three phases: planning, development and maintenance, and project scaling. In addition, the study highlights that careful planning, including risk assessment, change management and the creation of a center of excellence, is critical to ensure the sustainability of RPA initiatives. Assessing and monitoring aspects related to security, data management and governance are also key aspects to maintain the long-term viability of RPA solutions. While this study provides a valuable framework for the managerial and methodological sustainability of an RPA project, its focus is on governance and planning. It does not propose a specific technical architecture for integrating intelligent validation, which is the primary contribution of this study.

According to the study in [10], in Nordic companies the determinants for the implementation of Robotic Process Automation (RPA) as a digital transformation tool were explored. The technique used was qualitative in approach, using semi-structured interviews with experts from eight organizations in the medical, real estate and accounting sectors. The method combined case study, grounded theory and thematic content analysis. The results showed that the main motivation for adopting RPA was operational improvement through the automation of repetitive tasks, prioritizing labor cost savings. Although benefits such as increased productivity and freeing up time for higher-value tasks were observed, associated risks were also identified, such as technological dependence and vulnerability to system failures. On a quantitative level, the study reports that the return on investment (ROI) in the first year can vary between 30% and 200%, which demonstrates a significant financial impact on the companies analyzed. This research focuses on the business drivers and financial impact of adopting RPA, justifying why an organization would implement automation. However, it does not address the what or how of the implementation itself, specifically regarding the technical challenge of ensuring data quality. This study fills this gap by presenting a model that embeds ML validation directly into the automation workflow.

III. Concepts

This section presents the main concepts of this work. The objective is to develop a Robotic Process Automation (RPA) script that leverages Machine Learning techniques for enterprise data integration.

A. *Robotic Process Automation [11]*

It is an automation approach based on software bots that automate business processes by understanding existing procedures.

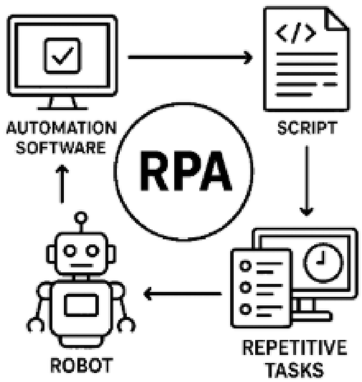

Fig. 1. Main components of this RPA scripting.

In Fig. 1, the main components of RPA, including automation software, scripts, digital robots, and repetitive tasks, and how they interact to optimize business processes is given.

B. *Enterprise Data Integration [12]*

A set of data extracted from different sources, i.e., raw data. These data require conversions and, after conversion, are stored through various technical processes, providing a single view of the data.

C. *Scripting [13]*

It is the creation and use of scripts which are programs or sets of instructions written and predefined in code made in some programming language to perform actions on computers in order to speed up the execution of processes and tasks.

D. *Python [14]*

High-level interpreted programming language created by Guido van Rossum in 1991.

E. *Python Libraries [15]*

Python libraries are tools that facilitate programming by providing predefined functions for various tasks. In automation and data analysis, Playwright for web browsing and Pandas for structured information processing stand out.

1) Playwright: Test automation framework for web applications, developed by Microsoft. It allows running tests in Chrome, Firefox, Safari, and Edge with a unified API, compatible with JavaScript, Python, and Java. It supports headless mode, network manipulation, and video recording to facilitate test analysis [16].

2) Pandas: Open-source software library designed for data manipulation and analysis in Python programming language [17].

3) Oracledb: It is a widely used database management system known for its robust features and tools for data design, management, and analysis. It supports object-oriented and relational data models, making it versatile for a variety of applications [18].

F. *Machine Learning [19]*

A subset of artificial intelligence (AI) that enables systems to learn from data and improve their performance over time without being explicitly programmed.

G. *Isolation Forest [20]*

The Isolation Forest (iForest) algorithm is a popular Machine Learning method for anomaly detection. It is particularly effective due to its low computational complexity and its ability to handle high-dimensional data.

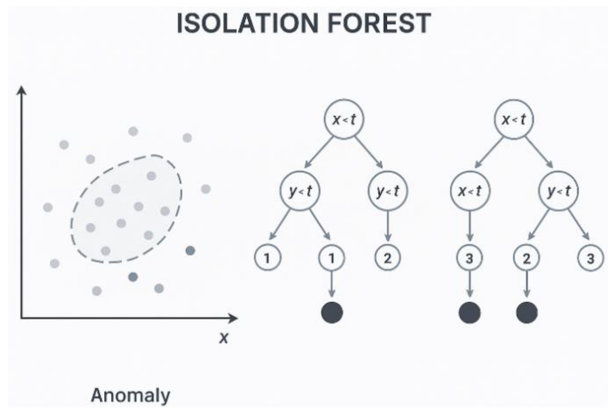

Fig. 2. Isolation forest.

Fig. 2 illustrates how the Isolation Forest algorithm isolates data points using multiple randomly generated trees. Anomalous points are isolated early and appear at shallower depths (dark nodes), whereas normal points require more splits (light nodes). The ensemble of these trees enables the systematic detection of anomalies.

H. *Random Forest [21]*

Versatile and widely used Machine Learning algorithm that combines multiple decision tree classifiers to improve predictive accuracy and control overfitting. It is especially effective for classification and regression tasks.

As shown in Fig. 3, the original data set is randomly divided into K subsets, and each is trained on an independent decision tree. Each tree makes its own prediction, and all the predictions are then combined by voting (in classification) or by averaging the results (in regression).

I. *Autoencoder [22]*

Neural network for unsupervised learning, used in dimensionality reduction and data reconstruction. It is composed of an encoder, which compresses the data into a latent space, and a decoder, which reconstructs it from this representation. The architecture of the autoencoder is shown in Fig. 4.

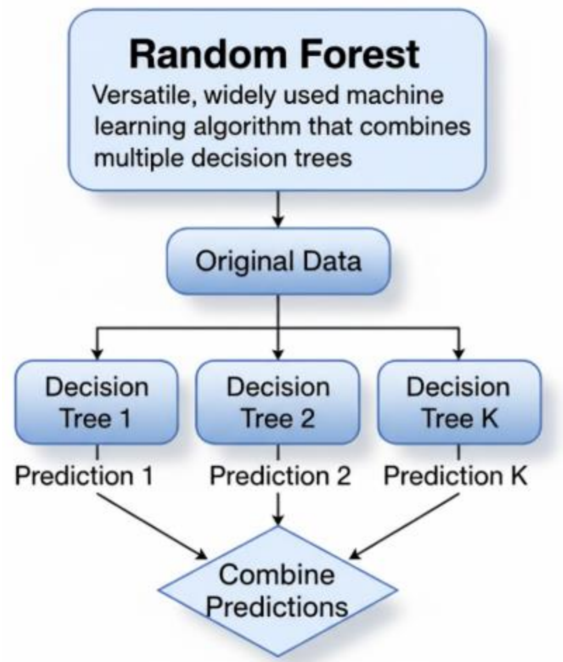

Fig. 3. Random forest.

As seen in Fig. 4, input data (such as daily patterns) is passed through the hidden layer, where the model compresses it into a smaller representation. The output layer then attempts to reconstruct the original data from that compressed representation. If the patterns are normal, the reconstruction will be accurate, but if it is an anomalous pattern, the reconstruction will be incorrect, which helps identify the anomaly.

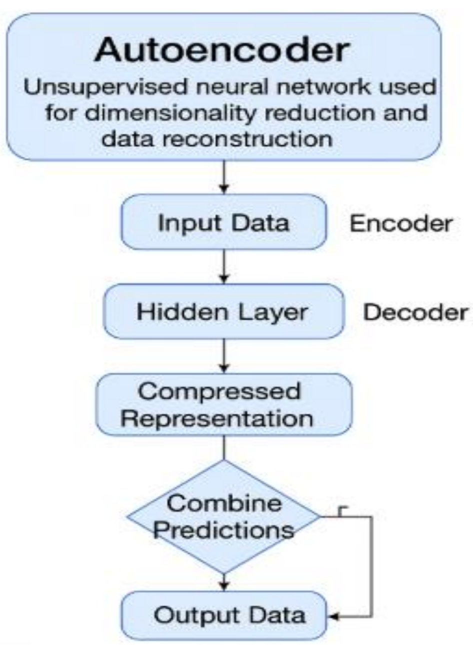

Fig. 4. Auto-encoder.

IV. MODEL

The proposed RPA model integrates specialized tools to intelligently automate the entire data integration flow. It all starts with the use of Playwright, which allows automated interaction with web platforms, performing tasks such as browsing, authentication, and file downloading in a robust and reliable way. Then, the extracted data is processed by Pandas, a library that allows the cleaning, transformation, and structuring of the information to ensure its quality and consistency. At this stage, a Machine Learning component is incorporated to strengthen data validation: Random Forest classifies records as valid or invalid when historical labels are available, Isolation Forest detects rows with subtle errors through unsupervised detection, and Autoencoders identify unusual deviations between daily reports, generating proactive alerts before proceeding. Once validated, the data is efficiently loaded into a relational database via Oracledb, ensuring its persistence and availability for analytical queries. Finally, the processed and stored information is visualized through data visualization tools, which allow building interactive dashboards for decision making based on reliable, timely, and automated audited data. The complete architecture of this model is shown in Fig. 5.

A. Data Capture and Extraction

In this first stage, Playwright is used, a modern and highly efficient tool that allows simulating human interactions with web browsers in a programmed way, which in this context is adapted to the RPA scripting paradigm to automate dynamic interaction with web platforms, allowing tasks such as navigation between pages, button clicks, credential entry, pop-up web window management, and file downloading. This automation ensures robust, reproducible, and human error-free data collection, laying the foundation for efficient downstream processing.

B. Intelligent Data Processing and Validation

In this stage, the extracted data undergoes a series of transformations using the Pandas library, an essential tool in the Python data analysis ecosystem. Here, cleaning tasks such as null removal, format normalization, field unification, creation of new derived columns, duplicate handling, and structure alignment are performed. However, the key differential of this stage is the incorporation of intelligent validation through Machine Learning algorithms, which allows not only verifying the structure but also detecting semantic or contextual errors in the data.

1) Random forest: Supervised algorithm trained with historical records labeled as valid or invalid, which automatically predicts whether a new record meets the logical integrity criteria or presents errors.

2) Isolation forest: An unsupervised algorithm that detects subtle anomalies in the data without the need for labels. This model analyzes the distribution and isolation of records to identify outliers.

3) Autoencoders: Neural network-based algorithm that detects unusual changes or structural deviations in daily reports compared to previous behavior, generating early warnings when anomalous patterns are identified, such as an atypical volume of data or unusual combinations of fields.

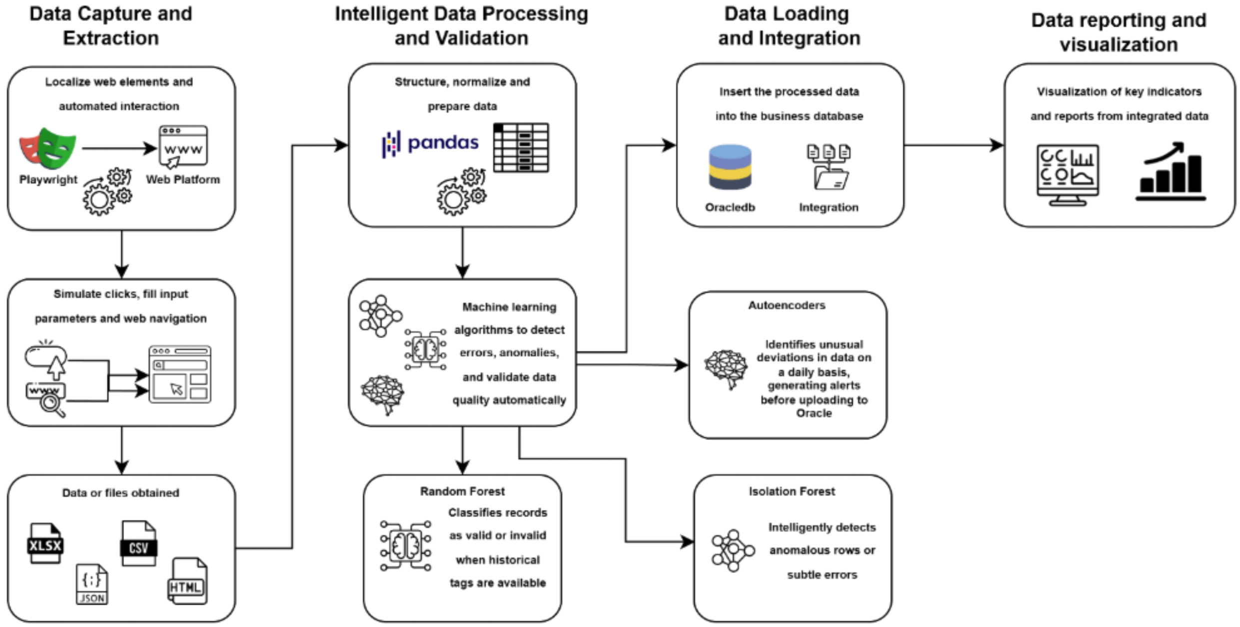

Fig. 5. RPA Scripting model using Machine Learning (ML) algorithms.

C. Data Loading and Integration

At this stage, validated and structured data is prepared for persistence in the corporate storage system. Using the Oracledb client, a secure and efficient connection is established with the Oracle database, where the controlled insertion of the records in the corresponding tables is performed, thus guaranteeing consistency, traceability, and availability of the information for the entire organization.

D. Reporting and Data Visualization

The last stage of the flow focuses on converting the processed data into actionable value. Here, data visualization tools are used (e.g., a Power BI dashboard showing campaign KPIs, anomaly alerts, and performance trends, as shown in Fig. 6) to connect directly to the Oracle database. This stage facilitates strategic analysis by presenting information in a clear, dynamic, and visual way, helping business teams to make decisions based on up-to-date, accurate data previously audited by the automated flow.

Fig. 6. Campaign performance dashboard.

V. RESULTS

To validate the proposed RPA model, its performance was evaluated in two key areas: operational efficiency in reducing execution time compared to manual workflows, and the effectiveness of the integrated Machine Learning algorithms for data validation.

A. Operational Performance Evaluation

First, the baseline reference times for the existing manual processes were established. As detailed in Table I, the manual processing of an average of 8,500 daily records required 3 hours (120 minutes), and the manual loading of 7,400 records required 4 hours (240 minutes).

TABLE I. DAILY OPERATIONAL BENCHMARK OF MANUAL FLOWS

Flows	Average daily amount of data records used	Average Manual Execution Times
Data processing	8500 daily records	3 hours
Data loading	7400 daily records	4 hours

Subsequently, the automated workflow was executed for the same tasks. Table II compares the manual versus automated execution times. Data processing was reduced from 120 minutes to 15 minutes, and data loading was reduced from 240 minutes to 40 minutes.

TABLE II. COMPARISON OF MANUAL AND AUTOMATED TIMES

Flow	Estimated time of execution of the manual process	Estimated time of execution of the automated process
Data processing	120 minutes	15 minutes
Data loading	240 minutes	40 minutes

To quantify the efficiency gain, Eq. (1) was used to calculate the percentage of time reduction (RT).

$$RT = \left(1 - \frac{T_{auto}}{T_{manual}}\right) * 100\% \quad (1)$$

where,

$$RT = Time\ reduction$$

$$T_{manual} = Manual\ execution\ time$$

$$T_{auto} = \text{Automated}\ execution\ time$$

In Eq. (1), the percentage of time reduction achieved by automating an operational process is calculated. It compares the average manual execution time with the automated time. The result indicates how much more efficient the process has become: a high RT value reflects a significant improvement in flow duration, allowing the impact of automation to be quantified in terms of time savings and operational optimization.

Now we apply Eq. (1) to calculate the percentage reduced after applying the automatic solution to both flows in Table I.

- For Data Processing:

$$RT = \left(1 - \frac{15}{120}\right) * 100\%$$

$$RT = 87.5\%$$

The result was that the reduction in time was 87.5%. That is, the 15 minutes represent 12.5% of 120 minutes.

- For Data Loading:

$$RT = \left(1 - \frac{40}{240}\right) * 100\%$$

$$RT = 83.33\%$$

The result showed a reduction in time of 83.33%. This indicates that the execution time has been reduced by that proportion by implementing automation. That is, 40 minutes represent 16.67% of 240 minutes.

B. Intelligent Validation Evaluation

In the second part of the evaluation, the performance of the Machine Learning algorithms (Autoencoder, Isolation Forest, and Random Forest) was measured based on their ability to validate and detect anomalies in the data.

Performance metrics, including Accuracy, Precision, Recall, F1-score, and training time, are presented in Table III.

TABLE III. VALIDATION METRICS FOR AUTOENCODER, ISOLATION FOREST, AND RANDOM FOREST

Algorithm	Accuracy	Precision	Recall	F1-score	Training time (s)
Isolation Forest	0.88	0.87	0.86	0.865	40
Random Forest	0.91	0.9	0.89	0.895	45
Autoencoder	0.92	0.91	0.9	0.905	70

As shown in Table III, the results from Table I were used to compare the performance of three algorithms: Autoencoder, Isolation Forest, and Random Forest. The Autoencoder performed best across all performance metrics (accuracy of 0.92 and F1-score of 0.905), albeit with a longer training time (70 s). The Random Forest showed a good balance between precision (0.90), recall (0.89), and speed (45 s), while the Isolation Forest was the fastest (40 s), but with slightly lower metrics (accuracy of 0.88 and F1-score of 0.865). These results demonstrate that each model exhibits distinct strengths depending on the validation objective.

VI. LIMITATIONS AND FUTURE WORK

This section details the limitations identified during the development of this research, which consequently establish the foundation for future lines of work.

A. Limitations

Although the model demonstrated significant improvements in operational efficiency and data validation accuracy, it is important to acknowledge several limitations:

1) Context-specific data: The model was trained and validated using a specific dataset: daily performance reports of marketing campaigns sent to users of a telecommunications company. The positive results, especially the ML validation metrics in Table III, are tied to this context. The model's applicability and performance in other business domains, such as finance, logistics, or human resources, are not guaranteed and would require separate validation.

2) Requirement for historical data: The model's validation process combines different learning methods. The Random Forest algorithm, which showed strong performance, is a supervised method. This means it requires a history of data already labeled as 'valid' or 'invalid' to be trained. In new implementations where this historical data is not available, the system must depend only on unsupervised methods (Isolation Forest and Autoencoder), which could affect the initial classification accuracy.

3) Error detection nuances: During testing, it was observed that while the Autoencoder was effective at finding structural deviations in daily reports, it sometimes struggled to interpret legitimate seasonal peaks in campaign performance, incorrectly flagging them as anomalies. This indicates a need for future models to better incorporate business seasonality.

4) Maintenance of RPA scripts: The model relies on Playwright for web data extraction. This creates a practical maintenance challenge. The automation scripts depend on web page elements (like CSS IDs) to function. If the source web platforms are updated or change their layout, these scripts can fail and will require manual updates to restore functionality.

B. Future Work

Based on the results obtained and the limitations described, future efforts will focus on improving the model's robustness, adaptability, and traceability:

1) Periodic algorithm retraining: It is recommended to implement a periodic retraining schedule for the Machine Learning algorithms. This is essential to maintain the model's accuracy as data patterns change over time. Updating the models with recent historical data ensures they remain effective and adaptive.

2) Enhanced traceability: A crucial next step is to implement systematic logging of key process events, such as data download, validation, insertion, and visualization. This practice not only facilitates audits and post-failure analysis but also supports the continuous improvement of the RPA model.

3) Formal version control: Finally, adopting a formal versioning policy for all models and scripts is highlighted as a priority. Maintaining version control for both the ML algorithms and the data extraction scripts allows for safe modifications and provides a mechanism to revert to a previous stable version if a failure occurs, significantly reducing operational risks.

VII. CONCLUSION

To begin with, the integration of Robotic Process Automation (RPA) Scripting with Machine Learning (ML) techniques offers a significant advantage over traditional automation approaches, which often lack intelligent validation and risk, propagating poor-quality data. The proposed model significantly enhances efficiency and accuracy by combining automated task execution with a multi-stage validation component using Random Forest, Isolation Forest, and Autoencoders. This layered approach provides a more robust defense against anomalies than research relying on a single algorithm. The system is capable of not only processing data autonomously but also identifying diverse types of errors and ensuring the reliability of the information before it reaches the database. This fusion reduces dependency on manual validation and mitigates data quality risks in dynamic, high-volume environments.

Moreover, the utilization of Pandas as the central tool for data manipulation enables a highly flexible and scalable processing layer. Its powerful functions for data cleansing, transformation, and preparation allow for seamless integration with ML models, ensuring that the input data meets quality standards before predictive or classification tasks are executed.

Finally, adopting an end-to-end automated architecture—from web data extraction to visualization—ensures complete traceability and transparency of the data lifecycle. This comprehensive approach empowers decision-makers with accurate and timely insights, presented through dynamic dashboards and reports. By aligning automation, validation, and visualization, the model offers a robust foundation for intelligent data-driven decisions, thereby establishing a sustainable and forward-looking solution for organizations aiming to scale their data operations efficiently.

REFERENCES

[1] M. Naeem, et al., "Trends and Future Perspective Challenges in Big Data," in *Advances in Intelligent Data Analysis and Applications*, J.S. Pan, V.E. Balas, C.M. Chen, Eds. Singapore: Springer, 2022, vol. 253. https://doi.org/10.1007/978-981-16-5036-9_30.

[2] R. K. Debbadi and O. Boateng, "Optimizing end-to-end business processes by integrating machine learning models with Uipath for predictive analyties and decision automation," *Int. J. Sci. Res. Arch*., vol. 14, no. 02, pp. 778-796, 2025. https://doi.org/10.30574/ijsra.2025.14.2.0448.

[3] H. Kong Journal and M. B. H. Kong, "Integrating Machine Learning-Driven RPA with Cloud-Based Data Warehousing for Real-Time Analytics and Business Intelligence," *Kong J. Al Med*., vol. 4.

[4] H. P. Kothandapani, "Integrating Robotic Process Automation and Machine Learning in Data Lakes for Automated Model Deployment, Retraining, and Data-Driven Decision Making," *Sage Sci. Rev. Appl. Mach. Learn*., vol. 4, no. 2, pp. 16-30, 2021. [Online]. Available: https://journals.sagescience.org/index.php/ssraml/article/view/167.

[5] P. William, S. Choubey, A. Choubey, and G.S. Chhabra, "Evolutionary Survey on Robotic Process Automation and Artificial Intelligence," in *Robotic Process Automation*, R. Rawat, et al., Eds., 2023. https://doi.org/10.1002/9781394166954.ch21.

[6] D. Kedziora and S. Hyrynsalmi, "Turning Robotic Process Automation onto Intelligent Automation with Machine Learning," in *Proc. 11th Int. Conf. Communities and Technologies (C&T '23)*, New York, NY, USA: Association for Computing Machinery, 2023, pp. 1-5. https://doi.org/10.1145/3593743.3593746.

[7] V. Jadkar, et al., "A Robotic Process Automation for Stock Selection Process and Price Prediction Model using Machine Learning Techniques," *Int. J. Recent Innov. Trends Comput. Commun*., vol. 10, no. 7, pp. 50-57, 2022. https://doi.org/10.17762/ijritec.v10i7.5569.

[8] S. Waduge, R. Sugathadasa, A. Piyatilake, and S. Nanayakkara, "A Process Analysis Framework to Adopt Intelligent Robotic Process Automation (IRPA) in Supply Chains," *Sustainability*, vol. 16, no. 22, p. 9753, 2024. https://doi.org/10.3390/su16229753.

[9] C. Daase, A. Pandey, D. Staegemann, and K. Turowski, "Sustainability in robotic process automation: Proposing a universal implementation model," in *Proc. 20th Int. Conf. Informatics in Control, Automation and Robotics (ICINCO)*, vol. 1, SciTePress, 2023, pp. 770-779. https://doi.org/10.5220/0012260200003543.

[10] D. Kedziora, A. Leivonen, W. Piotrowicz, and A. Öörni, "Robotic Process Automation (RPA) Implementation Drivers: Evidence of Selected Nordic Companies," *Issues Inf. Syst*., vol. 22, no. 2, pp. 21-40, 2021. https://doi.org/10.48009/2.iis.2021.21-40.

[11] D. A. d. S. Costa, H. S. Mamede, and M. Mira da Silva, "Robotic Process Automation (RPA) Adoption: A Systematic Literature Review," *Eng. Manag. Prod. Serv*., vol. 14, no. 2, pp. 1-12, 2022. https://doi.org/10.2478/emj-2022-0012.

[12] J. Sreemathy, et al., "Data Integration and ETL: A Theoretical Perspective," in *Proc. 7th Int. Conf. Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, India, 2021, pp. 1655-1660. https://doi.org/10.1109/ICACCS51430.2021.9441997.

[13] O. Oluwagbemi, et al., "An Analysis of Scripting Languages for Research in Applied Computing," in *Proc. 16th IEEE Int. Conf. Computational Science and Engineering*, Sydney, NSW, Australia, 2013, pp. 1174-1180. https://doi.org/10.1109/CSE.2013.174.

[14] T. Oliphant, "Python for scientific computing," *Comput. Sci. Eng*., vol. 9, no. 3, pp. 10-20, 2007. https://doi.org/10.1109/MCSE.2007.58.

[15] N. Pilnenskiy and I. Smetannikov, "Feature Selection Algorithms as One of the Python Data Analytical Tools," *Future Internet*, vol. 12, no. 3, p. 54, 2020. https://doi.org/10.3390/fi12030054.

[16] K. Brahmbhatt, "Comparative Analysis of Selecting a Test Automation Framework for an E-Commerce Website," Master's thesis, Inst. Comput. Syst., Tallinn Univ. Technol., 2023. [Online]. Available: https://digikogu.taltech.ee/et/Item/a5ef145c-0dcd-42c9-8e48-93943f8700e7.

[17] S. Hagedorn, S. Kläbe, and K. Sattler, "Putting Pandas in a Box," in *Proc. 11th Annu. Conf. Innovative Data Systems Research (CIDR)*, 2021. https://doi.org/10.22032/dbt.51534.

[18] L. He, D. Wang, S. Bao, and J. Yun, "Implementation of the object-oriented data model in Oracle database," *Appl. Mech. Mater*., vol. 44-47, pp. 3849-3853, 2011. https://doi.org/10.4028/www.scientific.net/AMM.44-47.3849.

[19] B. Datta and B. Sahoo, "Machine Learning, Regression and Optimization," in *Data Science and SDGs*, B.K. Sinha and M.N.H. Mollah, Eds. Singapore: Springer, 2021. https://doi.org/10.1007/978-981-16-1919-9_15.

[20] M. S. Lakshmi, et al., "Evaluating the Isolation Forest Method for Anomaly Detection in Software-Defined Networking Security," *J. Electr. Syst.*, vol. 19, no. 4, 2023. https://doi.org/10.52783/jes.639.

[21] Y. Liu, Y. Wang, and J. Zhang, "New machine learning algorithm: Random forest," *Lecture Notes in Computer Science*, vol. 7473 LNCS, pp. 246-252, 2012. https://doi.org/10.1007/978-3-642-34062-8_32.

[22] H. Motamednia, A. Mahmoudi, and A. Ng, "Autoencoders," in *Dimensionality Reduction in Machine Learning*, 2025, pp. 245-268. https://doi.org/10.1016/B978-0-44-332818-3.00020-4.