Mathieu Francois[1]
ORCID: 0009-0009-7808-771X

Iván Rivero[2]
ORCID: 0009-0006-0466-3065

Cristina Tirnauca[2]
ORCID: 0000-0002-7129-2237

Rafael Duque[2]
ORCID: 0000-0001-8636-3213

[1] ENSEIRB-MATMECA
Bordeaux INP, Avenue du Dr Albert Schweitzer, Talence, France

[2] Departamento de Matemáticas, Estadística y Computación
Universidad de Cantabria, Avda. de los Castros s/n, 39005 Santander, Spain

mathieu.francois@bordeaux-inp.fr, ivan.rivero@unican.es, cristina.tirnauca@unican.es, rafael.duque@unican.es

An Interactive AutoML and LLM-Based Platform for Medical Data Analysis

DOI: 10.34739/si.2025.33.01

Abstract. The widespread adoption of machine learning (ML) in healthcare is often limited by the significant technical expertise required to build, optimize and interpret predictive models. This paper presents an interactive platform designed to democratize access to medical data analysis by integrating an Automated Machine Learning

(AutoML) engine with a Large Language Model (LLM) conversational assistant. The proposed system enables non-technical users, such as clinicians and biomedical researchers, to upload datasets and generate robust predictive models using AutoGluon without the need for programming. To address the interpretability challenges of complex models, the platform couples SHAP-based visualizations with an LLM-driven chat interface that explains preprocessing steps, model metrics and feature importance in natural language. We evaluated the platform on eight public biomedical datasets covering both classification and regression tasks. Experimental results demonstrate that the system produces competitive predictive performance relative to established benchmarks while ensuring high usability. By lowering technical barriers and enhancing transparency, this tool empowers domain experts to independently leverage ML for clinical decision support and research.

Keywords: AutoML, LLM, Medical Data Analysis, Explainable AI, Human-Computer Interaction

1 Introduction

Artificial intelligence (AI) has become an essential component of modern scientific and technological development. In recent years, machine learning (ML) has moved from being a specialized research area to becoming a widely adopted tool for solving problems in many sectors. Organisations now rely on predictive modelling to support decision making, extract knowledge from large datasets and automate repetitive tasks that were traditionally performed manually [2]. This rapid growth has been fuelled by the increasing availability of digital data, improvements in algorithms and faster computing resources [1, 2]. As a result, many fields that previously lacked computational infrastructures have begun to integrate ML into their workflows in order to take advantage of its analytical capabilities.

Among all application areas, healthcare and biomedical research have experienced particularly significant transformations. Medical environments generate vast amounts of information, including laboratory records, clinical notes, imaging data, physiological sensor signals and genomic measurements. These data sources offer great potential for discovering clinical patterns, identifying risk factors and supporting diagnostic procedures. ML models can reveal relationships that are difficult to detect using traditional statistical methods and can assist practitioners in making more informed and timely decisions. The combination of diverse datasets with advanced algorithms has opened the door to personalised medicine and data-informed clinical strategies [6, 7, 13, 14].

Despite these opportunities, the adoption of ML in clinical practice faces important barriers. Developing a reliable predictive model usually requires a sequence of technical steps that include data cleaning, feature preprocessing, identification of suitable algorithms, model configuration and performance evaluation. Each of these tasks demands a certain level of expertise in data science and programming. Medical professionals, although highly specialised in their domain, typically do not have extensive training in these areas. As a consequence, they often need to collaborate with data scientists or software engineers, which introduces dependencies and delays in research workflows. This reliance also reduces flexibility, since clinicians may not be able to independently explore hypotheses or make iterative adjustments to the modelling process [2, 12].

Automated Machine Learning (AutoML) emerged as a response to these challenges. AutoML systems aim to reduce the technical burden by automating many steps of the ML pipeline. Over the years, these systems have evolved from simple hyperparameter optimizers to comprehensive frameworks capable of handling data preprocessing, model selection, ensembling strategies and complex search routines [3–5]. AutoML has made it possible for users with limited technical backgrounds to obtain strong predictive models without manual configuration [2]. However, despite its progress, AutoML still presents limitations. Most tools require users to understand concepts such as validation strategies, metric selection or model interpretability. In many cases, the defaults offered by AutoML systems may not be optimal for the specific characteristics of medical datasets, which are often noisy, heterogeneous and imbalanced. As highlighted in several studies, current AutoML solutions remain challenging for domain experts who do not have technical experience [6, 7, 12].

In parallel with these developments, large language models (LLMs) have brought a new dimension to human–computer interaction. LLMs are capable of understanding natural language, generating detailed explanations and reasoning about multi-step problems. Their ability to interpret free-text instructions provides an opportunity to design interfaces that feel more intuitive and accessible to non-technical users. In the context of ML, LLMs can clarify unfamiliar concepts, guide users through complex analyses and produce human-readable explanations of algorithmic behaviour. This makes them particularly valuable in domains such as medicine, where interpretability and clarity are essential for adoption [8, 11].

Integrating LLMs with AutoML offers a promising direction for democratizing ML. While AutoML systems automate model construction, LLMs can serve as an intelligent conversational layer that helps users understand the process, anticipate potential issues and interpret final results [10, 15, 16]. In medical contexts, this combination is especially relevant. Clinicians need to trust the systems they work with and must be able to justify how predictions were generated. A conversational assistant capable of explaining preprocessing decisions, describing feature importance and summarizing model limitations can substantially increase the transparency of the overall workflow. Moreover, the ability to interact through natural language reduces friction and gives researchers greater autonomy when exploring datasets [8].

The system presented in this work builds on these ideas. We introduce an interactive platform that integrates an AutoML engine with an LLM to support medical data analysis in an accessible and user-centred manner. The platform allows clinicians and biomedical researchers to upload tabular datasets, configure training settings using intuitive controls, generate predictive models and evaluate their performance. AutoGluon, an open-source AutoML framework, serves as the core automation engine, performing model exploration and optimization without requiring users to write code [4]. Interpretability is addressed through SHAP (SHapley Additive exPlanations) based visualizations that highlight the role of each variable in the model's predictions. The platform also incorporates an intelligent assistant powered by an LLM that provides explanations of the data, the preprocessing steps, the generated models and the obtained results. This conversational component helps users understand concepts that they may not be familiar with and bridges the knowledge gap between domain expertise and ML techniques.

The goal of this work is to create a tool that facilitates the use of ML in healthcare by prioritizing accessibility, interpretability and guidance. By combining automation with natural language support, the proposed platform empowers clinicians to take a more active role in

data-driven research and reduces their dependency on technical collaborators. At the same time, it offers a controlled workflow that avoids many common pitfalls associated with manual model development. The evaluation of the system on several biomedical datasets demonstrates that it is capable of producing competitive results while requiring minimal technical effort from the user [6, 13, 14]. Overall, this work contributes to the broader objective of making AI more usable and trustworthy in sensitive domains such as healthcare.

The remainder of this paper is organized as follows. Section 2 reviews related work on AutoML frameworks, their applications in healthcare and recent efforts to integrate AutoML with LLM-based conversational interfaces. Section 3 describes the architecture and implementation of the proposed platform. Section 4 presents the experimental evaluation conducted on several public biomedical datasets. Section 5 discusses the main limitations of the current system and outlines future research directions. Finally, Section 6 summarizes the main conclusions of this work.

2 Related Work

We outline key developments in AutoML frameworks, their applications in healthcare and recent trends combining AutoML with conversational LLM-based systems.

2.1 Advances in AutoML Frameworks

AutoML has matured significantly over the past decade, giving rise to robust frameworks that automate key steps of the ML pipeline. Early innovations in hyperparameter optimization, neural architecture search (NAS) and meta-learning have broadened AutoML beyond simple model selection [1, 2]. Modern open-source tools like Auto-sklearn, TPOT, Auto-Keras, H2O AutoML, AutoGluon and Auto-PyTorch integrate these advances to automatically handle feature engineering, model selection, model training and tuning. These systems have achieved near-expert performance on standard benchmarks by efficiently exploring model pipelines and hyperparameters [3–5] . Comparative evaluations indicate that no single AutoML tool is universally best; for example, a recent study benchmarking 16 frameworks found AutoGluon achieved the best overall balance of predictive accuracy and efficiency, while other tools excelled on specific tasks or metrics [4]. Such results underscore the trade-offs in AutoML design – some frameworks favour exhaustive search for maximum accuracy, whereas others prioritize speed or resource efficiency [1, 2]. Overall, the proliferation of AutoML libraries (including popular commercial platforms like Google Cloud AutoML and Microsoft's Azure AutoML) demonstrates the field's progress in reducing the technical overhead of model development.

Despite these successes, usability for non-technical users remains a core limitation of most AutoML frameworks. Notably, even the "user-friendly" AutoML platforms still require navigating complex configuration settings and understanding of ML [2]. In practice, domain experts without coding skills often struggle with these tools, as a basic understanding of ML (e.g., choosing model types, interpreting metrics) is needed to use them effectively [2]. For instance, AutoML systems like Auto-PyTorch reduce manual coding through high-level APIs, but their rigid presets force users to adapt to the tool's workflow, causing cognitive friction for

novices. In an in-depth survey, Santu et al. [12] observe that current AutoML pipelines still involve humans at crucial steps (problem formulation, data preparation, etc.), which keeps AutoML from being truly "automatic" for domain experts. In short, the literature suggests that most AutoML solutions were designed with algorithmic efficiency in mind over user-centric design, leaving a gap in accessibility for clinicians or other non-technical professionals.

2.2 AutoML Applications in Healthcare

The medical domain has become an important testing ground for AutoML, with applications in disease diagnosis, outcome prediction and medical imaging analysis. In medical imaging, Beduin et al. [13] developed AutoResCovidNet using AutoKeras to classify chest X-rays as healthy, pneumonia or COVID-19. Although the model did not outperform the best expert-designed alternatives, the study highlighted AutoML's value for rapid model development with limited human effort. Similarly, van Eeden et al. [14] evaluated AutoML for psychiatric diagnosis prediction and found that Auto-sklearn outperformed logistic regression and Naïve Bayes when using complex feature sets, while also showing more consistent accuracy across predictors. These studies suggest that AutoML can simplify modelling for clinicians while remaining competitive on complex healthcare tasks.

Beyond individual case studies, broader evaluations have also shown both the potential and the limitations of AutoML in healthcare. Romero et al. [6] compared Auto-sklearn, H2O and TPOT on highly imbalanced insurance-claims datasets for disease prediction. All three outperformed a hand-tuned Random Forest baseline, although none was consistently superior across all tasks. This suggests that AutoML performance in healthcare remains task-dependent, particularly under class imbalance. Another major challenge is interpretability, since AutoML often produces complex models that are difficult to explain. A systematic review by Yuan et al. [7] found that most healthcare AutoML models still lack sufficient interpretability, which limits clinical adoption. To address this issue, recent work has explored integrating explainability techniques such as feature importance and rule extraction into AutoML pipelines, seeking to combine automation with greater transparency and trust in clinical settings.

2.3 Integrating AutoML with Conversational LLM Interfaces

Recent work has begun to fuse AutoML with LLMs to create interactive, conversational interfaces for AI model building. Several researchers propose an "LLM-as-Translator" paradigm, where an LLM interprets the user's free-text instructions and converts them into AutoML pipeline configurations or code [8, 9]. This line of work includes systems like AutoML-GPT [10], which employs a GPT-based model as a bridge between user requests and ML operations. In AutoML-GPT, a user can describe a task (for example, "predict which patients are at risk of diabetes from these health records") and the GPT agent will autonomously generate the entire ML pipeline (from data preprocessing and feature engineering to model selection and hyperparameter tuning) and execute it. Other contemporary efforts [15, 16] have similarly used LLMs to generate model training code or API calls for popular AutoML frameworks based on natural-language descriptions. All these initiatives share the goal of

democratizing AI, allowing domain experts to build sophisticated models through dialogue instead of writing code.

Initial evidence is promising that LLM-integrated AutoML can significantly lower barriers for non-programmers. Yao et al. [11] conducted a human-subject study comparing a conversational AutoML system (with an LLM interface) against traditional AutoML tooling and manual coding. Remarkably, 93% of users in the study achieved equal or higher model accuracy using the LLM-based interface than they did with a code-based approach, while over 60% of users also reported substantially faster completion times for the same tasks. The natural language interface was especially beneficial for participants with little ML experience: it helped them complete complex classification tasks that they would have struggled to implement via code, effectively bridging the skill gap. These results underscore the potential of conversational interfaces to improve both the effectiveness and efficiency of AutoML, by guiding users through model development in a more intuitive way. Importantly, the study also noted fewer errors and a faster learning curve when users interacted with the LLM assistant, suggesting that such interfaces can serve as educational tools in addition to automating workflow steps [11].

That said, integrating LLMs with AutoML is not without challenges. One notable issue is the "circular dependency" of requiring some ML knowledge to use even a natural language AutoML assistant effectively [2, 12]. For instance, AutoML-GPT's interface allows plain English input, but users still must know certain domain-specific terminology (e.g., the difference between "classification" vs. "regression", or the notion of an "evaluation metric") to formulate requests that the system can correctly interpret [10]. If a medical user is unfamiliar with these concepts, they may struggle to get useful results from the system – essentially the user needs ML expertise to fully benefit from a tool that is supposed to obviate the need for ML expertise [8, 12]. Researchers are beginning to address this by improving prompt design and interactive guidance. For example, Guo et al. [16] introduced an LLM-driven AutoML platform that attempts to clarify assumptions and ask follow-up questions, reducing the user's need to manually specify technical details. Nevertheless, rigorous evaluations of such systems are still limited. Most existing studies focus on the technical viability of LLM-driven pipelines rather than comprehensive user experience across diverse real-world tasks [10, 15, 16]. Moving forward, a key research direction is to refine conversational AutoML systems so that they can truly act as intelligent assistants, handling ambiguity in user instructions, explaining the modeling choices and requiring minimal prerequisite knowledge. The related work surveyed above lays the foundation for our proposed platform, which aims to build on these ideas by tightly coupling an AutoML engine with an LLM-based conversational interface, thereby democratizing AI access for non-experts users by removing the barriers of traditional programming.

3 System Architecture and Implementation

The system enables the construction and interpretation of predictive models from structured datasets by users without prior training in programming or ML. It is implemented as a self-contained application that can be downloaded and executed locally. Users only need a standard Python environment to launch the tool, which starts a local server and

opens an interface in the browser. This browser interface functions purely as a viewing and interaction layer, while all computation, preprocessing and model training are carried out entirely on the user's machine. A public demonstration of the system is available at https://huggingface.co/spaces/ivanriza99/Demo_app, allowing users to explore the workflow without installing anything. The demo runs exclusively on CPU and therefore does not reflect the performance benefits available on machines equipped with a GPU. Since the conversational assistant relies on an external LLM service, users of the demo must also provide their own Groq API key to enable LLM-based explanations and guidance.

3.1 General Architecture and Workflow

The system follows a modular design in which the interface, the AutoML engine and the conversational assistant work together as a unified workflow. Once the application is executed, users can upload a dataset, configure preprocessing settings and launch automated model training within a guided environment. A typical workflow unfolds as follows:

1. **Dataset loading and inspection:** Users upload a dataset in CSV, Excel or ARFF format. The interface immediately displays a preview of the data and summary statistics to verify correctness and quality.
2. **Preprocessing configuration:** Users may remove unnecessary variables, select strategies for handling missing values and choose the target variable for prediction.
3. **Automated model training:** The application invokes AutoGluon, which internally manages preprocessing, model exploration, hyperparameter optimization and validation. The user only specifies a training time limit.
4. **Result visualization:** After training, the system presents performance metrics, model leaderboards and graphical diagnostics such as confusion matrices, ROC curves or residual plots. SHAP-based visualizations offer insights into feature importance.
5. **Conversational guidance:** At any moment, users can interact with the integrated language model assistant, powered by Groq, to obtain explanations, clarifications or recommendations about preprocessing, metrics or model behaviour.
6. **Model reuse and prediction:** Trained models can be downloaded or reloaded within the application. Users can upload new datasets and generate predictions, which can be previewed and exported.

This workflow allows researchers to build and interpret ML models without writing code or configuring complex tools. Screenshots of the steps appear in Appendix **??**.

3.2 System Components

The application is structured into several functional components that operate locally:

– **Web-based interface:** Built with HTML, CSS and JavaScript, it provides panels for dataset inspection, preprocessing options, model training, visualization of results and interaction with the chat assistant.

– **Local Python backend:** The server is implemented using **Flask**, a lightweight Python framework that facilitates RESTful API development and seamless integration with ML workflows. It is responsible for all processing, including data handling, invocation of AutoGluon, generation of evaluation plots and management of trained models.
– **Language model assistant:** A conversational component accessed via the Groq inference API that explains technical concepts, interprets model outputs and guides iterative refinement. By default, the system integrates *meta-llama/llama-4-scout-17b-16e-instruct* but users may also choose other LLMs provided through Groq to suit their specific needs. To enable context-aware responses, the system serializes the AutoML results (including performance metrics, confusion matrices and feature importance scores) into a structured JSON summary. This summary is injected into the system prompt, allowing the LLM to ground its answers in the specific empirical results of the user's current session rather than providing generic definitions.
– **Local file management:** Datasets, models and generated artifacts are stored in temporary directories on the user's machine. Files are cleared as needed, minimizing storage usage and ensuring data remains under the user's control.

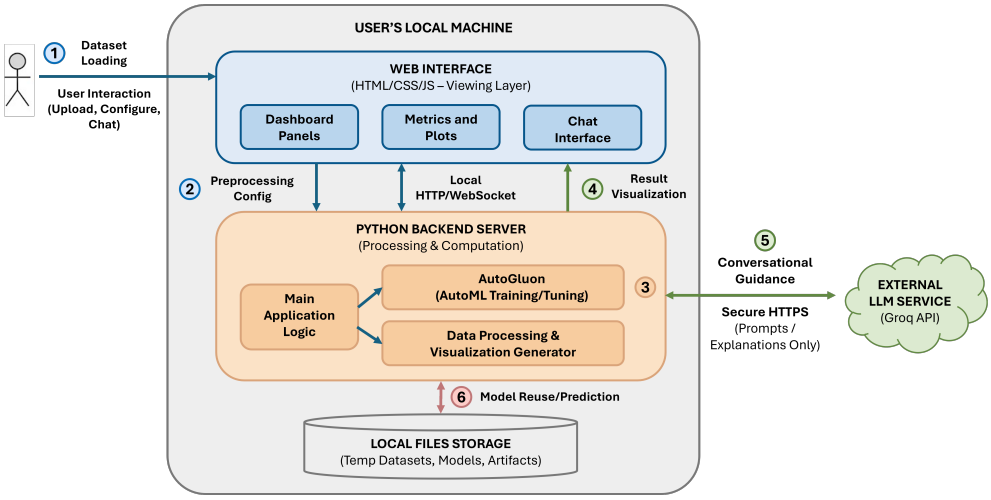


Figure 1: System architecture overview

3.3 Practical Considerations and Efficiency

Several implementation decisions enhance the practicality of the platform:

– **Privacy-preserving local execution:** A critical requirement in healthcare is the protection of sensitive patient data. By performing all model training and inference locally on the user's device, the platform ensures that raw clinical data never leaves the local environment. This architecture inherently aligns with strict data protection regulations (such as GDPR or HIPAA) by design, removing the risks associated with uploading medical records to third-party cloud AutoML services.

– **Controlled resource usage:** AutoGluon employs mechanisms such as early stopping and adaptive search, enabling strong performance even on modest hardware.
– **Automatic housekeeping:** Temporary files are periodically cleaned to maintain responsiveness and limit disk usage.
– **Lightweight distribution:** Running the system requires only installing Python dependencies. No additional configuration or infrastructure is needed.

4 Experimental Evaluation

To assess the effectiveness of the proposed platform, we conducted a series of experiments on publicly available medical datasets involving both classification and regression tasks (see Figure 2). The goal of the evaluation was not only to measure predictive performance, but also to verify whether the platform could produce strong baseline models with minimal user intervention, reflecting its intended use by clinicians and non-technical researchers.

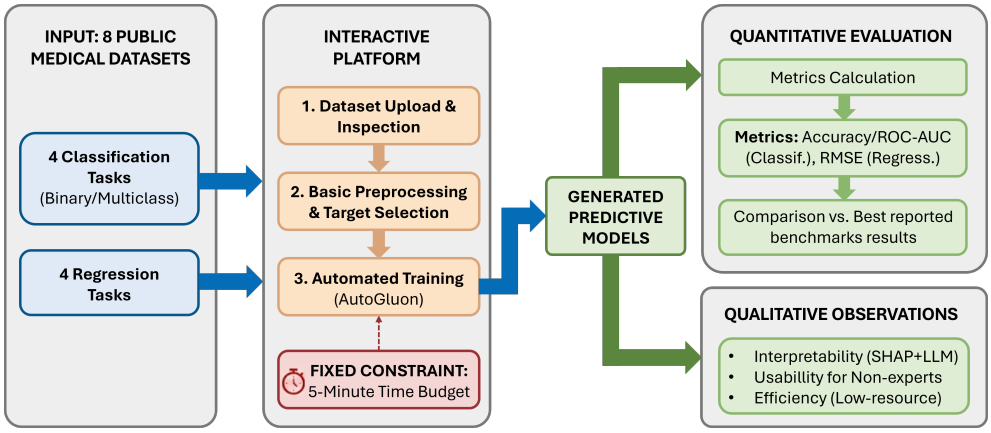


Figure 2: Experimental evaluation workflow.

4.1 Experimental Setup

All experiments were executed locally on a standard workstation using the downloadable version of the system. Each dataset was processed through the complete workflow of the platform: dataset upload, basic preprocessing, selection of the target variable and automated training with AutoGluon. Unless otherwise specified, the default settings of the system were preserved to simulate realistic usage conditions for non-expert users.

A fixed time budget of five minutes was allocated for AutoML exploration in all experiments. This constraint ensures comparability across datasets while reflecting a practical training duration suitable for interactive sessions. For classification tasks, the primary evaluation metrics were accuracy and ROC-AUC. For regression tasks, we used root mean squared error (RMSE). These metrics were selected because they are widely reported in the public leaderboards of the corresponding datasets, enabling fair comparisons.

4.2 Datasets

Eight publicly available biomedical datasets were selected from platforms such as Kaggle and OpenML. These datasets span different clinical domains and present a variety of modelling challenges, including class imbalance, heterogeneous feature types and limited sample sizes. Four datasets correspond to classification tasks (binary or multiclass) and four correspond to regression tasks. All datasets include predefined train–test splits, which were used to ensure consistent evaluation and to avoid data leakage.

Table 1 summarizes their source, task type, train/test sizes and dimensionality. For OpenML datasets, both the Dataset ID and Task ID are provided.

Table 1: Summary of the eight biomedical datasets used in this study. For OpenML datasets, Dataset ID and Task ID are included.

Dataset	Source	Task	Train size	Test size	Features
Breast Cancer (Wisconsin)	OpenML (ID 15, Task 245)	Classification (2)	469	230	9
Pima Indians Diabetes	OpenML (ID 37, Task 267)	Classification (2)	515	253	8
Contraceptive M. C.	OpenML (ID 23, Task 253)	Classification (3)	987	486	9
Hypothyroid	OpenML (ID 57, Task 3044)	Classification (3)	2528	1245	29
Liver Disorders	OpenML (ID 8, Task 211690)	Regression	232	113	5
BrisT1D Blood Glucose	Kaggle	Regression	177024	3644	506
COVID-19 Death Prediction	Kaggle	Regression	129156	43052	18
COVID-19 Cases Prediction	Kaggle	Regression	2700	893	92

4.3 Results

Table 2 summarizes the performance obtained by our platform compared with the best publicly reported scores for each dataset. Although the goal of the system is not to outperform highly optimized competition-level models, the results provide a benchmark for assessing the quality of models produced under realistic, low-intervention conditions.

Table 2: Performance comparison on eight public medical datasets.

Dataset	Task	Metric	Our Result	Best Reported
Breast Cancer (Wisconsin)	Binary classification	ROC-AUC	0.980	0.991
Pima Indians Diabetes	Binary classification	ROC-AUC	0.785	0.841
Contraceptive Method Choice	Multiclass classification	Accuracy	0.565	0.574
Hypothyroid	Multiclass classification	Accuracy	0.997	0.998
Liver Disorders	Regression	RMSE	3.35	3.38
BrisT1D Blood Glucose	Regression	RMSE	2.56	2.36
COVID-19 Death Prediction	Regression	RMSE	299	226
COVID-19 Case Prediction	Regression	RMSE	1.036	0.869

The platform consistently produced competitive models across all datasets. In several cases, including Hypothyroid and Liver Disorders, the performance was nearly identical to the best leaderboard results. For more complex tasks, such as COVID-19 case and mortality prediction, a larger gap was observed. These differences can be attributed to the limited training

time and the fact that top leaderboard entries often employ extensive feature engineering, domain-specific preprocessing or ensemble strategies tuned explicitly for competition settings.

Nevertheless, the system achieved results that are more than adequate for exploratory modelling and early-stage research, especially considering that the models were generated with minimal user input and no manual hyperparameter tuning.

4.4 Qualitative Observations

Beyond numerical performance, several qualitative observations emerged during experimentation:

- **Robustness across domains:** The platform handled heterogeneous datasets without requiring manual adjustments, demonstrating the reliability of the automated preprocessing and modelling pipeline.
- **Interpretability of results:** The combination of AutoGluon's native feature importance plots and SHAP-based visualizations consistently provided meaningful explanations of feature influence, which the conversational assistant successfully translated into plain language. However, it was observed that generating these interpretability artifacts, particularly SHAP values, can be extremely computationally expensive for high-dimensional datasets or those with a large number of instances (consequently, the application treats the generation of these specific visualizations as optional to preserve workflow efficiency).
- **Efficiency:** Training times remained within the predefined limits and allowed for an interactive workflow. Even on modest hardware, AutoGluon produced models with strong predictive performance.
- **Usability for non-experts:** The presence of the LLM assistant significantly improved the interpretability of complex outputs, making model evaluation more accessible to users with limited ML background.

Overall, the experimental evaluation demonstrates that the platform is capable of producing high-quality predictive models, offering competitive performance relative to publicly reported benchmarks. The tool balances automation, interpretability and usability, validating its suitability as a practical solution for clinicians and researchers who wish to integrate machine learning into their workflows without requiring extensive technical expertise.

5 Limitations and Future Directions

Although the proposed platform demonstrates strong potential for enabling non-experts to build and interpret machine learning models, several limitations must be acknowledged. These limitations highlight opportunities for further development and guide future research aimed at strengthening the system's clinical relevance and technical robustness.

5.1 Current Limitations

First, the platform currently focuses exclusively on structured tabular data. Many medical datasets include additional modalities such as imaging, free-text clinical notes or longitudinal

time-series, which are not yet supported. Extending the system beyond tabular inputs would broaden its applicability to more complex clinical scenarios.

Second, the automated modelling pipeline relies on a fixed time budget and a predefined set of AutoGluon configurations. While this design simplifies interaction for non-technical users, it restricts more advanced users who may wish to explore model families, adjust optimization strategies or fine-tune hyperparameters. Providing optional expert-level controls without compromising usability remains a challenge.

Third, the system depends on an external API for LLM-based guidance. Although no private data are transmitted and prompts are carefully filtered, reliance on an online service may limit the use of the platform in highly sensitive environments or in institutions with strict network policies. An offline or on-premise LLM option would mitigate this limitation and enhance privacy assurance.

Fourth, the evaluation presented in this study focuses on publicly available datasets. These datasets are useful for benchmarking but do not reflect the full complexity of real-world clinical data, which often include missing data, schema inconsistencies or domain-specific biases. More comprehensive validation with genuine clinical datasets and domain experts is necessary to confirm the system's performance and usability in practical settings.

Finally, the current conversational assistant provides explanations on demand but does not yet support multi-turn analytical guidance, proactive error detection or tailored adaptation to user expertise. An adaptive assistant able to recognize user objectives, anticipate misunderstandings and propose corrective actions would further improve usability, especially for medical professionals with limited ML background.

5.2 Future Directions

Several extensions are planned to address these limitations. Future versions of the platform will explore the integration of multimodal data, including support for medical imaging, time-series signals and free-text documents. This expansion would allow the system to tackle common clinical tasks such as radiological classification, patient monitoring or clinical note summarization.

Another important direction is the incorporation of more flexible AutoML controls. Providing an "advanced mode" with optional configuration panels would allow experienced users to guide the search space or apply domain-specific constraints while maintaining the simplicity required for beginners.

On the LLM side, we aim to explore offline or locally hosted language models. This would remove the dependency on external services and facilitate deployment in hospitals or research institutions where data privacy is critical. In addition, more sophisticated conversational workflows will be developed, enabling the assistant to track analysis context, perform reasoning over intermediate results and support iterative refinement of models.

We also plan to conduct user-centred evaluations with clinicians and biomedical researchers. These studies will assess usability, cognitive load, trust in model explanations and the practical value of the tool in real decision-making workflows. Feedback from these evaluations will inform interface design, explanation strategies and model validation requirements.

Finally, we intend to extend the reporting capabilities of the system to include automated audit trails, reproducibility artifacts and compliance-oriented documentation. Such additions

would support the integration of AutoML workflows into clinical pipelines that require traceability and regulatory alignment.

6 Conclusions

This work presents an interactive platform that combines AutoML and LLM guidance to make data analysis more accessible to non-expert users. By integrating AutoGluon for automated model development with a conversational assistant capable of explaining concepts, interpreting results and supporting decision-making, the system lowers the technical barriers that often prevent clinicians and biomedical researchers from engaging with ML tools.

The platform provides an end-to-end workflow that includes dataset inspection, preprocessing, automated training, interpretability through SHAP-based visualizations and reusable model deployment, all within a user-friendly local interface. Experimental results on a diverse set of public biomedical datasets demonstrate that the system can generate competitive predictive models with minimal manual intervention. In addition, qualitative observations highlight the value of the conversational assistant in clarifying model behavior and guiding users who may have limited ML experience.

Beyond predictive performance, the platform emphasizes usability, transparency and educational value. These qualities are essential for fostering trust in AI-assisted clinical workflows and for enabling domain experts to explore data-driven hypotheses independently. The public demo further illustrates the system's accessibility and provides a practical entry point for new users.

While the current implementation focuses on tabular datasets and relies on an external LLM API, the architecture is flexible and can be extended to integrate new modalities, advanced AutoML configurations or offline conversational models. Future developments will focus on expanding multimodal capabilities, enhancing adaptability to user expertise and conducting user studies with medical professionals.

Acknowledgements This work has been partially supported by grants PID2022-139237NB-I00 and PID2023-146243OBI00 funded by MICIU AEI/10.13039/501100011033 and by ERDF/EU.

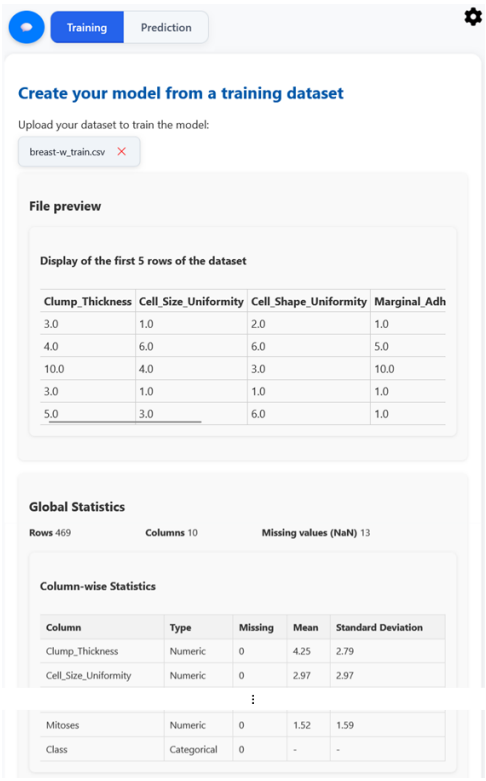
References

1. Hutter, F., Kotthoff, L., Vanschoren, J. (eds.): Automated Machine Learning: Methods, Systems, Challenges. Springer, Cham (2019).
2. He, X., Zhao, K., Chu, X.: AutoML: A survey of the state-of-the-art. Knowl.-Based Syst. **212**, 106622 (2021). https://doi.org/10.1016/j.knosys.2020.106622
3. Feurer, M., Klein, A., Eggensperger, K., Springenberg, J.T., Blum, M., Hutter, F.: Efficient and Robust Automated Machine Learning. In: Advances in Neural Information Processing Systems 28 (NeurIPS 2015), pp. 2962–2970 (2015).
4. Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., Smola, A.: AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data. arXiv:2003.06505 (2020).

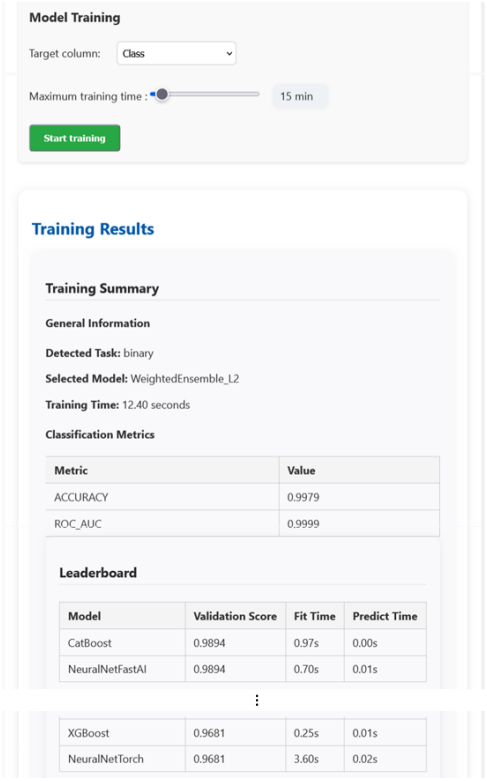
5. Olson, R.S., Moore, J.H.: TPOT: A Tree-based Pipeline Optimization Tool for Automating Machine Learning. In: Hutter, F., Kotthoff, L., Vanschoren, J. (eds.) Automated Machine Learning, pp. 151–160. Springer, Cham (2019).
6. Romero, R.A.A., Deypalan, A.R., Jungao, J.T.: Benchmarking AutoML frameworks for disease prediction using medical claims. BioData Min. **15**(1), 15 (2022). https://doi.org/10.1186/s13040-022-00300-2
7. Yuan, H., Yu, K., Xie, F., Liu, M., Sun, S.: Automated machine learning with interpretation: A systematic review of methodologies and applications in healthcare. Med. Adv. **3**(1), 42–55 (2024). https://doi.org/10.1002/med4.75
8. Tayebi Arasteh, S., Han, T., Lotfinia, M., et al.: Large language models streamline automated machine learning for clinical studies. Nat. Commun. **15**, 1603 (2024). https://doi.org/10.1038/s41467-024-45879-8
9. Miguel N., Rivero I., García D., Duque R., Palazuelos C. and Casas A. . Integrating Large Language Models into Automated Machine Learning: A Human-Centric Approach. In Proceedings of the 17th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management - Volume 1: KDIR; SciTePress, pages 465-472 (2025). https://doi.org/10.5220/0013819700004000
10. Zhang, S., Gong, C., Wu, L., Liu, X., Zhou, M.: AutoML-GPT: Automatic Machine Learning with GPT. CoRR abs/2305.02499 (2023).
11. Yao, J., Zhang, L., Huang, J.: Evaluation of large language model-driven AutoML in data and model management from human-centered perspective. Front. Artif. Intell. **8**, 1590105 (2025). https://doi.org/10.3389/frai.2025.1590105
12. Santu, S.K.K., Hassan, M.M., Smith, M.J., Xu, L., Zhai, C., Veeramachaneni, K.: AutoML to Date and Beyond: Challenges and Opportunities. ACM Comput. Surv. **54**(8), Article 175 (2022). https://doi.org/10.1145/3470918
13. Beduin, I.R.O.: Detecção da Covid-19 em imagens de raio-x: construindo um novo modelo de aprendizado profundo utilizando AutoML. Trabalho de Conclusão de Curso, Universidade de Brasília, Faculdade de Tecnologia (2021).
14. van Eeden, W.A., Luo, C., van Hemert, A.M., Carlier, I.V.E., Penninx, B.W., Wardenaar, K.J., Hoos, H., Giltay, E.J.: Predicting the 9-year course of mood and anxiety disorders with automated machine learning: A comparison between auto-sklearn, naïve Bayes classifier and traditional logistic regression. Psychiatry Res. **299**, 113823 (2021). https://doi.org/10.1016/j.psychres.2021.113823
15. Liu, S., Gao, C., Li, Y.: Large Language Model Agent for Hyper-Parameter Optimization. arXiv preprint arXiv:2402.01881 (2024).
16. Guo, J., Chen, Z., Ji, Y., Zhang, L., Luo, D., Li, Z., Shen, Y.: UniAutoML: A Human-Centered Framework for Unified Discriminative and Generative AutoML with Large Language Models. arXiv preprint arXiv:2410.12841 (2024).

Appendix A: User Interface Screenshots

This appendix contains screenshots of the system's user interface, shown in separate panels to illustrate the workflow across data upload, preprocessing, model training, interpretability and interaction with the LLM assistant, using the training partition of the Breast Cancer Wisconsin dataset.

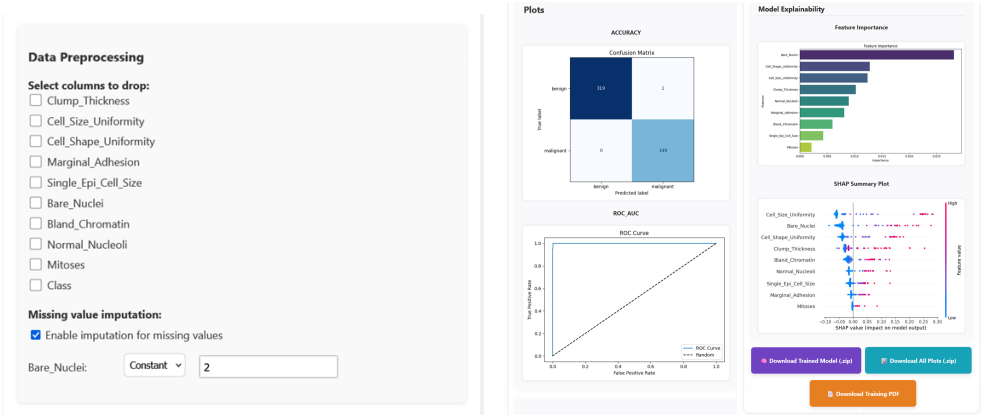


(a) Part I: dataset upload and statistics.



(b) Part III: training configuration and summary.

Figure 3: These panels correspond to different parts of the same workflow screen, shown separately for readability.



(a) Part II: manual preprocessing options.

(b) Part IV: interpretability and outputs.

Figure 4: Panels from the same workflow screen, shown separately for readability.

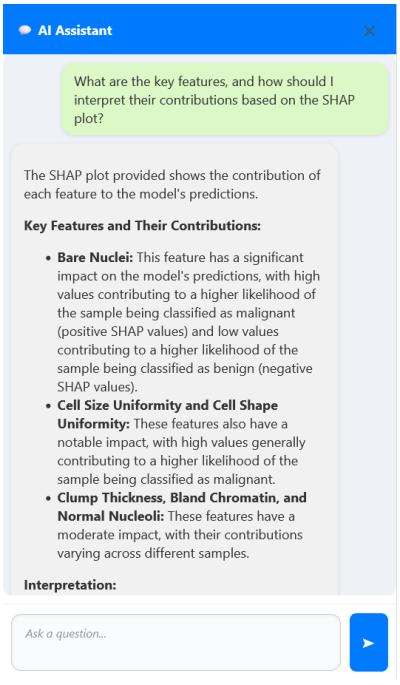


Figure 5: Part V: chat with the LLM.