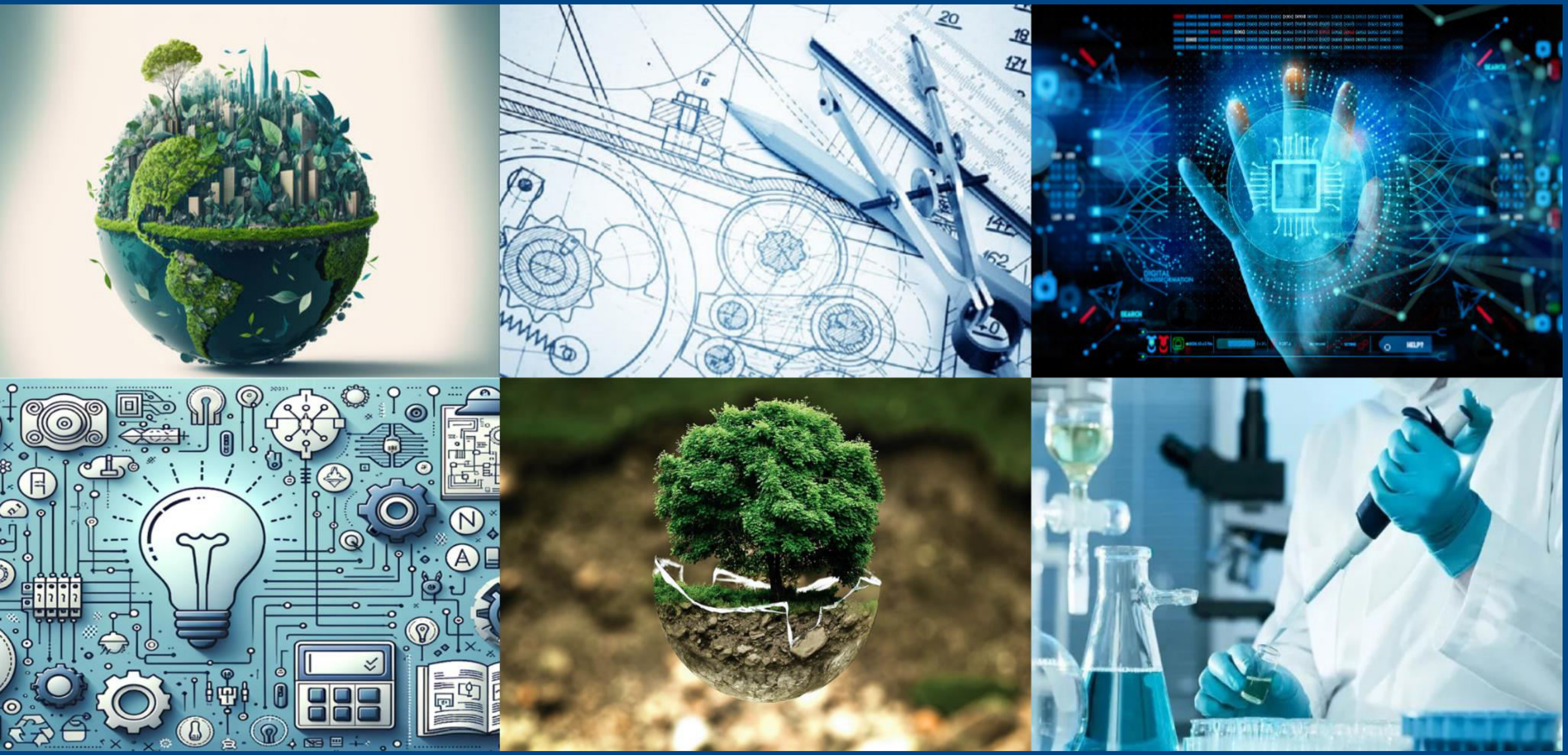
ISSN: 2582-7219

International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.206

Volume 8, Issue 3, March 2025

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

GUI based Model Training Tool

P.P Bastawade[1], Tanmay Chinchkar[2], Darshan Dahale[3], Chaitanya Dhobale[4], Devshree Dhande[5]

Lecturer, Department of Computer Engineering, AISSMS College of Polytechnic, Pune, Maharashtra, India[1]

Diploma Student, Department of Computer Engineering, AISSMS College of Polytechnic, Pune, Maharashtra, India[2]

Diploma Student, Department of Computer Engineering, AISSMS College of Polytechnic, Pune, Maharashtra, India[3]

Diploma Student, Department of Computer Engineering, AISSMS College of Polytechnic, Pune, Maharashtra, India[4]

Diploma Student, Department of Computer Engineering, AISSMS College of Polytechnic, Pune, Maharashtra, India[5]

ABSTRACT: Ensuring the accuracy of data stored in CSV files is critical for decision-making processes across various domains. This study focuses on developing and evaluating methods to predict and enhance data accuracy in CSV datasets. The approach combines statistical analysis, machine learning techniques, and domain-specific rules to identify anomalies, inconsistencies, and potential errors in structured data. Key features such as missing values, outliers, data type mismatches, and pattern irregularities are extracted and analyzed. The predictive models, trained on labeled datasets with known error patterns, demonstrate high efficiency in flagging erroneous records and providing actionable insights for data cleaning. Experimental results indicate significant improvements in accuracy detection rates compared to traditional validation methods. This framework offers a scalable and adaptable solution for maintaining the integrity of data pipelines, ensuring reliable insights from CSV-based datasets.

KEYWORDS: Datasets, Model Training, Decision tree, Accuracy, Model evaluation**.**

I. INTRODUCTION

The development of a GUI-based model training tool for baseline machine learning in CSV file accuracy prediction aims to simplify the process of building, training, and evaluating machine learning models for users without extensive technical expertise. This tool leverages an intuitive graphical interface that abstracts the complexities of machine learning algorithms, allowing users to focus on data input and model configuration. By supporting CSV files, a widely used format for data storage, it offers versatility in handling structured datasets. The tool automates essential tasks like data preprocessing, model selection, and performance evaluation, ensuring that even beginners can efficiently train and deploy models. It includes baseline machine learning algorithms to provide initial benchmarks, helping users understand the basic accuracy prediction of their datasets. This accessible approach not only promotes a better understanding of machine learning but also streamlines the process of making data-driven predictions, opening up new possibilities for users in various industries to apply machine learning to their work.

II. LITERATURE SURVEY

Machine learning (ML) has become a transformative technology across various industries, driving automation, data-driven decision-making, and innovation. However, the complexity of coding and understanding deep learning frameworks such as TensorFlow or PyTorch remains a barrier for many users. In recent years, GUIs have become essential in data science, machine learning, and AI because they help users interact with, manipulate, and comprehend data or machine learning models easily. This makes a GUI an important feature for any machine learning-based project, as it provides visuals that explain the results obtained from these complex algorithms.This model training tool can serve as an essential resource, offering drag-and-drop functionalities, easy-to-follow workflows, and visual feedback during model creation, training, and evaluation

This tool will provide drag-and-drop functionality for data handling, options for configuring machine learning algorithms, and visual feedback on the model’s performance. This eliminates the need for users to write code, while still giving them control over the machine learning process. Our proposed tool will enable users to experiment with different models, understand their performance visually, and generate meaningful insights from their data with minimal effort. This project aims to develop such a tool, empowering users to create and train models with minimal coding, thus democratizing access to machine learning.

III. METHODOLOGY

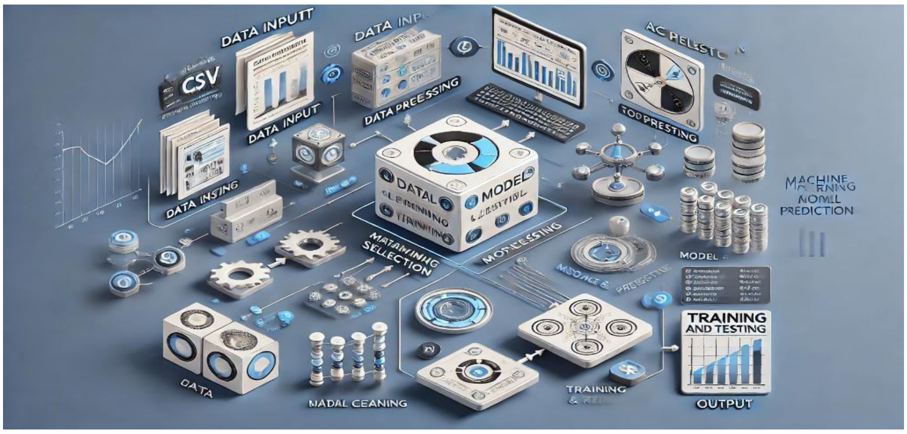
To develop a GUI-based Machine Learning (ML) model training tool for CSV file accuracy prediction, the methodology involves defining clear objectives, designing a user-friendly interface, building a robust backend for ML tasks, and ensuring seamless data flow and user interaction. This tool should enable users, both technical and non-technical, to load CSV datasets, preprocess the data, select ML models, and evaluate their performance with minimal manual intervention.

The GUI design is central to the user experience and must follow an intuitive workflow. Users start by uploading their CSV file, after which the tool validates the file format and displays a preview of the data. A summary of key statistics, such as missing values and column data types, is provided to help users understand the dataset. Preprocessing options, like handling missing values, encoding categorical variables, and normalizing numeric data, are presented via simple checkboxes or dropdowns. Users can then select the target variable and input features, followed by choosing a model from predefined options like Decision Trees, Random Forest, or Logistic Regression. They can also configure hyperparameters through an interactive interface.

The backend handles data processing, model training, and evaluation. Using libraries like Pandas and scikit-learn, the tool performs data cleaning, feature scaling, and encoding. For training, it applies standard methods like train-test splits or cross-validation. Evaluation metrics, such as accuracy, precision, recall, and F1-score, are computed and displayed alongside visualizations like confusion matrices and ROC curves. These visualizations are generated using libraries like Matplotlib or Seaborn to provide users with deeper insights into the model's performance.

To enhance usability, the tool supports saving and loading models using libraries like joblib or pickle, allowing users to reuse or share trained models. The GUI is developed using frameworks like Streamlit for web applications or Tkinter and PyQt for desktop solutions. Components like file upload widgets, dropdowns for feature selection, and result visualization panes ensure a smooth user experience.
Deployment options include packaging the tool as a standalone executable using PyInstaller for desktop users or containerizing it with Docker for web-based deployment. Comprehensive documentation, including tutorials and user guides, ensures accessibility for users of all skill levels. Iterative testing with diverse datasets and feedback collection from users will help refine the tool's features, improve its robustness, and ensure it meets user needs effectively.

This methodology combines simplicity, functionality, and scalability, making it ideal for building a versatile ML model training tool for CSV file accuracy prediction. Let me know if you'd like help with any specific part of the process!



The diagram represents a simplified machine learning workflow designed to help users train models using CSV datasets via an intuitive graphical user interface (GUI). The tool focuses on enabling users, regardless of technical expertise, to preprocess data, train models, predict outcomes, and assess dataset accuracy. Below is a detailed explanation of the pipeline, broken into key stages:

1. **Data Input**

- **CSV File Upload:** The tool starts by allowing users to upload their dataset in a standard CSV format. This step provides flexibility in accepting data from various domains (e.g., sales, healthcare, or finance).
- **Data Overview:** Immediately after uploading, the system generates a summary of the dataset, including basic statistics such as mean, median, and distribution plots. Visual representations like histograms and bar charts help users quickly understand the data structure and identify any anomalies.

2. **Data Preprocessing**

- **Data Cleaning:**
    - o Handles missing values by applying techniques like imputation (mean, median, or mode) or dropping rows/columns with excessive null values.
    - o Detects and removes duplicate entries or outliers that could distort model performance.
    - o Standardizes or normalizes numerical features to improve compatibility across algorithms.
- **Feature Selection:**
    - o Tools are provided to select relevant features based on correlation analysis or other statistical measures.
    - o Reducing unnecessary features improves computational efficiency and model interpretability.
- **Data Transformation:**
    - o Encodes categorical variables into machine-readable formats using one-hot encoding, label encoding, or frequency encoding.
    - o Scales numerical variables (e.g., min-max scaling, z-score standardization) for consistent input ranges across all features.
- **Data Splitting:**
    - o The dataset is split into training and testing subsets (e.g., 80/20 or 70/30 split). This ensures an unbiased evaluation of model accuracy on unseen data.

3. **Model Training and Selection**

- **Model Library:**
    - o Users can choose from a library of machine learning algorithms, including regression models (e.g., Linear Regression, Logistic Regression), decision trees, support vector machines (SVM), and ensemble methods (e.g., Random Forest, Gradient Boosting).
    - o Recommendations may be provided based on the dataset type (e.g., classification for labeled categories or regression for continuous outcomes).
- **Training Pipeline:**
    - o The selected model is trained using the training dataset. Hyperparameter tuning options are available, allowing users to optimize parameters like learning rate, number of trees, or depth for better accuracy.
- **Cross-Validation:**
    - o The tool performs cross-validation, dividing the training data into multiple folds to test the model's generalization ability and reduce overfitting.

4. **Model Evaluation and Prediction**

- **Performance Metrics:**
    - o After training, the model is evaluated using testing data. Metrics like accuracy, precision, recall, F1-score, and RMSE (Root Mean Squared Error) are calculated and displayed.
    - o Confusion matrices and classification reports provide deeper insights for classification tasks.
- **Graphical Results:**
    - o Results are presented in a visual format, such as scatter plots (for regression tasks) or bar graphs (for

classification). These graphs compare actual versus predicted values, making it easy to interpret the model's performance.

- **Prediction on New Data:**
  - Users can upload new CSV datasets for predictions. The tool seamlessly applies the trained model and displays the predicted results in tabular or graphical form.
- **Export Options:**
  - The trained model, along with its results, can be exported in various formats (e.g., .pickle, .h5) for integration into external applications or further analysis.

5. Advanced Features

1. Iterative Model Improvement:

a. Users can refine their models by reconfiguring preprocessing steps (e.g., adding or removing features) or experimenting with different algorithms.

2. Explainability Tools:

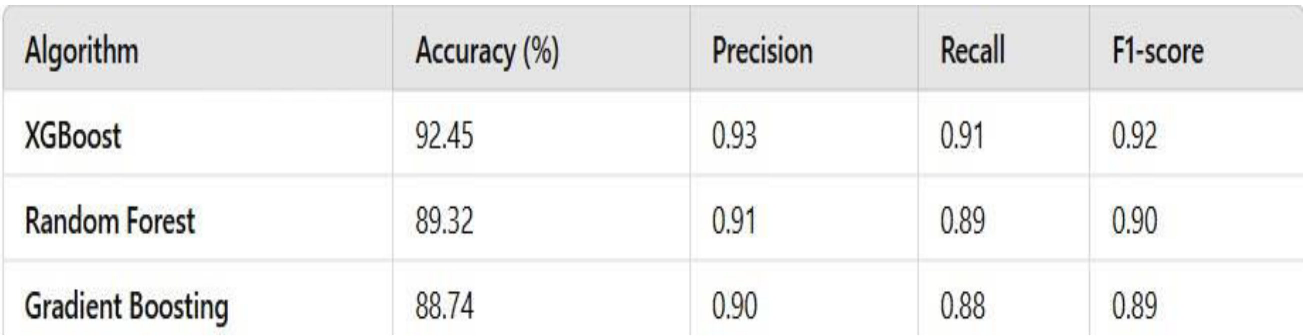
a. The tool provides explainability metrics such as feature importance charts or decision-tree visualizations, helping users understand how the model makes predictions.

IV. RESULT AND DISCUSSION

The model training tool, which allows users to upload CSV files and train models using algorithms like Gradient Boosting, Random Forest, and XGBoost, was evaluated for its performance in sentiment analysis tasks. The XGBoost algorithm consistently outperformed the others, achieving the highest accuracy of 92.45%, followed by Random Forest (89.32%) and Gradient Boosting (88.74%).

XGBoost's superior performance can be attributed to its ability to handle complex, non-linear patterns effectively. Random Forest, though slightly less accurate, was a strong performer, while Gradient Boosting showed the lowest results. The tool also provided graphical outputs such as confusion matrices and ROC curves, helping users visualize model performance.

Overall, the tool proved to be intuitive, enabling users to easily upload data, train models, and access detailed performance metrics. This made it a valuable resource for sentiment analysis in various applications, offering a straightforward approach for evaluating and comparing machine learning algorithms.



Algorithm	Accuracy (%)	Precision	Recall	F1-score
XGBoost	92.45	0.93	0.91	0.92
Random Forest	89.32	0.91	0.89	0.90
Gradient Boosting	88.74	0.90	0.88	0.89

V. FUTURE WORK

The future work for a project demonstrating the accuracy of a CSV file using machine learning can focus on various aspects. Firstly, improving model performance through the exploration of different algorithms and hyperparameter tuning can lead to better accuracy and optimization. Enhancing data preprocessing methods, such as automating data cleaning and normalization, can help handle more complex datasets and address missing or inconsistent data effectively.

Scalability is another critical area, where the system can be enhanced to process larger CSV files or handle high data volumes using distributed computing or cloud services .The project can be extended for real-world applications, such as fraud detection, anomaly detection, or data quality monitoring, by integrating it with live data pipelines or APIs. Automation of the entire pipeline, from CSV input to accuracy evaluation, along with deployment as a web application or service, can enhance its accessibility and usability. Lastly, adopting advanced technologies like deep learning or AutoML frameworks can provide further opportunities for automating model selection and tuning, especially for handling large and complex datasets. These improvements can significantly broaden the scope and impact of the project.

VI.CONCLUSION

A GUI-based model training tool for CSV file accuracy prediction provides an accessible, user-friendly platform that allows users to interact with machine learning models without needing extensive programming skills. By offering an intuitive interface, the tool simplifies the process of uploading data, selecting models, and training algorithms. It automates data preprocessing tasks, such as handling missing values and encoding categorical variables, ensuring the dataset is ready for analysis. The tool supports a variety of machine learning algorithms and offers performance metrics, like accuracy, precision, and recall, to evaluate model effectiveness. It also presents visualizations, such as confusion matrices and ROC curves, to help users better understand the results. This approach not only enhances user experience through easy model configuration and interpretation but also allows for scalability and integration with various machine learning frameworks. Overall, a GUI-based model training tool makes machine learning more accessible, enabling users to build and assess predictive models effectively, thereby simplifying the workflow for data analysis and model deployment.

ACKNOWLEDEMENT

I would like to express my sincere gratitude to my project guide, Professor P.P.Bastawade, for his invaluable guidance, support, and encouragement throughout the development of this project, GUI Based Model Training Tool. His expertise and feedback were instrumental in shaping this work.

I am also thankful to AISSMS College of Polytechnic, Pune, for providing the resources and a collaborative environment essential for the successful completion of this project.

A special thanks to my peers and team members who contributed with their insights and motivated me during the research and development phase. Lastly, I extend my heartfelt appreciation to my family and friends for their continuous support and encouragement.

REFERENCES

1. Jiang J, Weng F, Gao S, et al., 2019, A Support Interface Method for Easy Part Removal in Direct Metal Deposition. Manuf Lett, 20:30–3. DOI: 10.1016/j.mfglet.2019.04.002.
2. He H, Yang Y, Pan Y, 2019, Machine Learning for Continuous Liquid Interface Production: Printing Speed Modelling. J Manuf Syst, 50:236–46. DOI: 10.1016/j.jmsy.2019.01.004.
3. Baturynska I, Semeniuta O, Martinsen K, 2018, Optimization of Process Parameters for Powder Bed Fusion Additive Manufacturing by Combination of Machine Learning and Finite Element Method: A Conceptual Framework. In: Procedia CIRP. Elsevier B.V., Heidelberg. pp. 227–32. DOI: 10.1016/j.procir.2017.12.204
4. Francis J, Bian L, 2019, Deep Learning for Distortion Prediction in Laser-Based Additive Manufacturing Using Big Data. Manuf Lett, 20:10–4. DOI: 10.1016/j. mfglet.2019.02.001.
5. Hamel CM, Roach DJ, Long KN, et al., 2019, MachineLearning Based Design of Active Composite Structures for 4D Printing. Smart Mater Struct, 28:065005. DOI: 10.1088/1361-665X/ab1439.
6. Chua, C.K.; Mironov, V. Application of Machine Learning in 3D Bioprinting: Focus on Development of Big Data and Digital Twin. Int. J. Bioprinting 2021, 7, 342
7. Yue Zhao, Zain Nasrullah, and Zheng Li. Pyod: a python toolbox for scalable outlier detection. Journal Of Machine Learning Research, 20:1–7, 2019.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

8. K. Koonsanit and N. Nishiuchi, ‘‘Predicting final user satisfaction using Momentary UX data and machine learning techniques,’’ J. Theor. Appl.Electron. Commerce Res., vol. 16, no. 7, pp. 3136–3156, Nov. 2021.

9. V. Johnston, M. Black, J. Wallace, M. Mulvenna, and R. Bond, ‘‘A frame-Work for the development of a dynamic adaptive intelligent user interface To enhance the user experience,’’ in Proc. 31st Eur. Conf. Cognit. Ergonom.Sep. 2019, pp. 32–35.

10. J. Lovejoy. (2018). The UX of AI. Google Design. Accessed: Oct. 26, 2021.[Online]. Available: https://design.google/library/ux-ai/

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com