Identification of Algorithm Based On the Given Dataset Using AI-ML Techniques

D. Kavitha [1], P. Sony[2], D. Manvitha rani[3], A. Upendhar [4]
[1]*Assistant professor, Dept of IT, TKR College of Engineering and Technology.*
[2,3,4] *Student, Dept of IT, TKR College of Engineering and Technology.*

Abstract- **Today, as data continues to reach immense proportions (such as those generated by social media, smart devices, and digital platforms), traditional modes of data analysis tend to be inaccurate and slow. This project leverages artificial intelligence (AI) and machine learning (ML) to address one of big data's greatest challenges that is selecting the proper algorithm for a given dataset.**
Rather than guess and check, the system examines features like size and structure, and data types of the dataset to determine what type of problem it is whether classification, regression3, clustering, or anomaly detection. It recommends an algorithm with the combination of the exploration process described above, so that the selection of the best algorithm is more efficient.
The project includes tools for data processing, finding patterns, building models, and measuring their performance. It also lets users compare different algorithms.

I.INTRODUCTION

In today's world, we are surrounded by vast amounts of data. Processing and analyzing this data is essential for decision-making across fields. Artificial Intelligence (AI) and Machine Learning (ML) provide powerful tools for detecting patterns, making predictions, and solving complex problems.
Choosing the right machine learning algorithm for a given dataset can be a real challenge. Each dataset is performance and make data work easier unique, and choosing the correct method requires much trial and error, manual labor, and domain expertise. This can render the process both time-consuming and frustrating at times.

This project aims to automate that whole process. Using AI and machine learning algorithms, the system processes the dataset to learn about its structure and automatically identifies the problem type whether it's classification, regression, or clustering. From this analysis, it suggests the most appropriate algorithms and handles important steps such as data preprocessing, feature selection, and model evaluation.

The aim is to speed up, improve the intelligence of, and make algorithm choice more dependable. Rather than having to depend on so much experience or guesswork, this system makes educated decisions to enhance and more accessible to everyone.

II. LITERATURE SURVEY

2.1 Title: A Comparative Study of Machine Learning Algorithms for Classification.
Author: A. M. Alshamrani.
Description: This study analyzes various machine learning algorithms for classification tasks. It evaluates their performance on multiple datasets, focusing on accuracy and efficiency. By comparing the strengths and weaknesses of different models, the research highlights how they handle various dataset types. It also explores the trade-offs between computational complexity and predictive accuracy, offering guidance on selecting the best algorithm for specific classification problems.

2.2 Title: Selecting Appropriate Machine Learning Algorithms.
Author: P.M. DeLima.
Description: This study conducts a systematic literature review to propose a framework for selecting machine learning algorithms based on dataset characteristics. The framework emphasizes the identification of dataset features—such as size, structure, and the nature of the data (categorical, numerical, or mixed)—to guide the algorithm selection process. By aligning dataset properties with algorithm strengths, the study aims to optimize performance for various tasks and ensure that the selected algorithms are well-suited to the problem at hand. This approach facilitates an informed and structured decision-making process for both practitioners and researchers.

2.3 Title: Performance Analysis of Regression Algorithms.
Author: J. Smith.
Description: This study focuses on a comparative performance analysis of regression algorithms. It employs both simulation techniques and real-world datasets to evaluate the efficiency and reliability of various regression methods. The objective is to assess the performance of these algorithms when applied to different datasets, providing valuable insights into their effectiveness for diverse applications.

2.4 Title: Challenges in Unsupervised Learning Algorithm Selection.
Author: R. Brown.
Description: This study provides a comprehensive review of the challenges associated with selecting unsupervised learning algorithms. It examines the existing literature on unsupervised learning techniques and incorporates case studies and algorithm comparisons to offer practical insights. The research aims to explore the complexities and limitations in choosing appropriate algorithms for unsupervised tasks, focusing on both theoretical and applied perspectives.

2.5 Title: Hyperparameter Tuning in Machine Learning.
Author: K. J. Thakur.
Description: This work presents a review of various hyperparameter tuning methods and their effects on machine learning algorithms. The primary focus is on analyzing how hyperparameter tuning can enhance the performance of machine learning models. It explores practical tuning strategies while emphasizing the importance of optimizing hyperparameters to achieve higher model accuracy.

III. METHODOLOGY

1: User Interface and File Upload:
- The application begins with a minimal web interface through which users can upload a dataset in CSV format.
- There is a dropdown menu in the interface for the selection of problem type (Classification, Regression, Clustering).
- On submission, the file and the selected type are passed to the backend for processing.

2: Backend Routing and Input Validation:
- The Flask application routes two primary routes: displays the home page, predict supports POST requests to process the uploaded dataset.
- The uploaded file is checked, read with pandas, and the type of problem is determined or confirmed from user input.

3: Data Preprocessing:
- The dataset is processed by the backend system.
- Preprocessing involves: Dropping missing values and duplicates, Label encoding for categorical features, Standard scaling of numerical attributes
- When classification is selected and the data set is unbalanced, SMOTE is employed for class balancing.

4: Training Model Depending Upon the Problem Type: Depending on the chosen type, a specific processing function is called:
- process_classification()
- process_regression()
- process_clustering()

Classification:
- Trains models like SVC, Random Forest, Logistic Regression, Naive Bayes, etc.
- Tests them based on accuracy score.

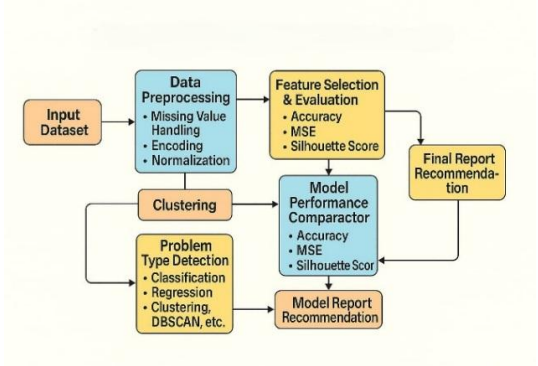
Regression:
- Trains models such as SVR, Linear Regression, Random Forest Regressor, etc.
- Tested using R^2 Score and Mean Squared Error (MSE).

Clustering:
- Executes using unsupervised models such as KMeans, DBSCAN, Gaussian Mixture, etc.
- Evaluates using Silhouette Score to determine cluster quality.

5: Performance Comparison and Visualization:
- The scores/accuracy of all models trained are saved and compared.
- A bar graph is generated using matplotlib to visualize performance.
- Automatically, the best-performing model is detected.

6: Result Display: The following are displayed:
- Best model name
- Model performance chart
- Tabular score comparison
- The user can choose to go back and test another dataset.

IV. SYSTEM ARCHITECTURE



The system has a smarter and more comprehensive method of operating with data. It starts by pre-processing the data—filling in missing values, converting text to numbers, and scaling values to a common scale.

It also employs clustering to group similar data points, which aids in the discovery of underlying patterns, particularly in unlabeled datasets, and aids in the exposure of patterns and relationships that would remain unseen otherwise. This stage adds another layer of awareness, which is especially useful in cases like fraud identification, customer profiling, or anomaly detection.

Prior to training any models, the system examines the data in order to discover the most informative features and digs deeper into patterns through visualizations. This helps in improving accuracy and understanding of the models.

The system then trains a variety of models for classification, regression, and clustering. Rather than just selecting the top one, it demonstrates how each of the models fared so that users can simply compare them.

It employs the appropriate performance measures for every task—such as accuracy for classification, MSE for regression, and silhouette score for clustering—to provide an overall picture of how well every model performs.

In addition to all this, the system also learns from historical results. Over time, it improves at suggesting the appropriate algorithm for a particular dataset, resulting in faster, more accurate decisions, and fully data-driven in the future.

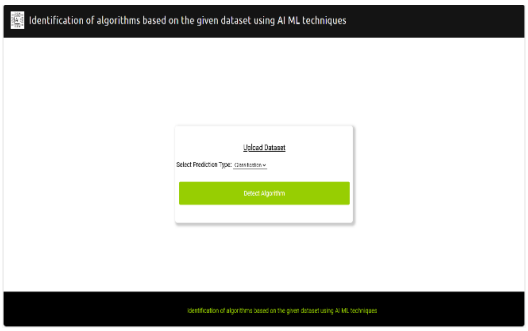
ADVANTAGES

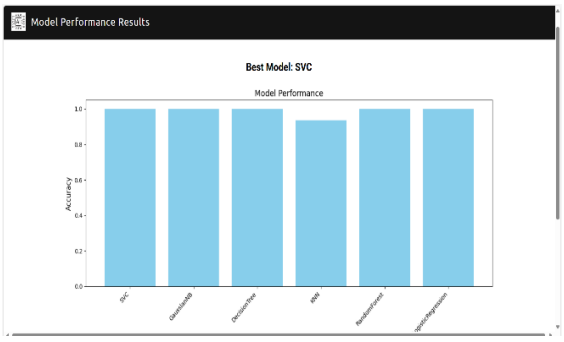
1. Rather than only indicating the best model, it indicates how each algorithm performed, so it is simpler to see the results overall.
2. Clustering also facilitates easier detection of anomalies—outliers stand out more when normal data is clustered together.
3. With clustering integrated, the system is now able to handle unlabeled datasets, assisting users in identifying useful patterns without predefined outputs.

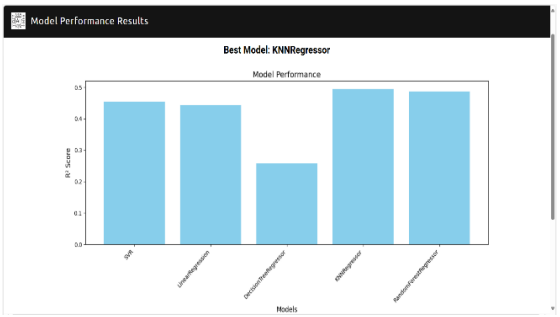
V. FUTURE SCOPE

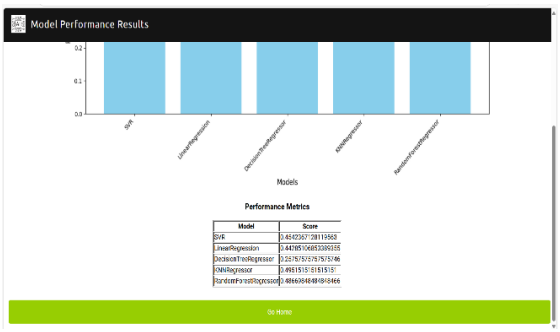
Looking ahead, this project has the potential to grow in many exciting directions. Right now, it does a great job of helping users choose the best machine learning algorithm based on their dataset, but there's so much more it could do. For example, it could be upgraded to include AutoML tools that take care of not just model selection but also fine-tuning and optimization — saving even more time and effort. It can also be extended to work with deep learning models, making it suitable for tasks involving images, videos, or audio files. Another major improvement would be adding explainable AI features that help users understand why a certain model was chosen or why it made a particular prediction, which is especially useful in areas like healthcare or finance. Hosting the system on cloud platforms and making it accessible through APIs would allow companies to easily integrate it into their existing workflows. There's also potential to make it work with real-time data streams, which would be a big step forward for applications like fraud detection or live monitoring. In the future, the system could also learn from user feedback to get even better at recommending models over time. Features like combining multiple models to improve accuracy, tracking past performance, and allowing users to log in and view personalized suggestions would make it even more powerful. And with a more interactive, mobile-friendly interface and options to export trained models, it could become a go-to tool for students, researchers, and professionals alike.

VI.RESULT









VII.CONCLUSION

In this increasingly data-driven, fast-paced world we're in today, the right machine learning approach shouldn't be a wild guess. Such a system, as it has been created, should behave similarly to a sagacious assistant a one that knows the structure of your data, identifies what kind of problem you're attempting to solve, and recommends the appropriate algorithms for you automatically. It does the heavy lifting: from data preprocessing and cleaning to comparing multiple models side by side and even giving insights into why one performs better than the others.

What makes this system stand out is its ability to handle not just labeled data for prediction, but also unlabeled data in order to find hidden patterns by clustering. This makes it extremely useful in actual applications like fraud detection, customer segmentation, or identifying unusual trends all without deep technical know-how.

By giving a full picture of model performance and learning from the past history of results, the system improves over time. It turns complicated machine learning procedures into a simple, insightful, and productive exercise. Whether you are a seasoned individual or a beginner, this system puts smart data exploration in your hands making machine learning not only more powerful, but also more accessible to humans.

REFERENCES

[1] Alshamrani, A. M. (2020). *A Comparative Study of Machine Learning Algorithms for Classification*. International Journal of Advanced Computer Science and Applications, 11(3), 123–130.

[2] DeLima, P. M. (2019). *Selecting Appropriate Machine Learning Algorithms Based on Dataset Characteristics*. Journal of Data Science and Machine Learning, 7(2), 101–113.

[3] Smith, J. (2018). *Performance Analysis of Regression Algorithms Using Real-World Data*. International Journal of Computational Intelligence, 9(1), 55–68.

[4] Brown, R. (2021). *Challenges in Unsupervised Learning Algorithm Selection*. Journal of Machine Intelligence, 18(2), 77–90.

[5] Thakur, K. J. (2022). *Hyperparameter Tuning in Machine Learning: A Practical Approach*. Journal of Artificial Intelligence Applications, 13(4), 301–316.

[6] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830.

[7] Lemaître, G., Nogueira, F., & Aridas, C. K. (2017). *Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning*. Journal of Machine Learning Research, 18(17), 1–5.

[8] Jain, A. K. (2010). *Data Clustering: 50 Years Beyond K-Means*. Pattern Recognition Letters, 31(8), 651–666.
[9] Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*. KDD, 226–231.
[10] Zhang, T., Ramakrishnan, R., & Livny, M. (1996). *BIRCH: An Efficient Data Clustering Method for Very Large Databases*. ACM SIGMOD Record, 25(2), 103–114.
[11] Raschka, S. (2015). *Python Machine Learning*. Packt Publishing.
[12] Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.). O'Reilly Media.
[13] Han, J., Kamber, M., & Pei, J. (2011). *Data Mining: Concepts and Techniques* (3rd ed.). Morgan Kaufmann.
[14] Buitinck, L., et al. (2013). *API Design for Machine Learning Software: Experiences from the Scikit-learn Project*. ECML PKDD Workshop.