www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

Automated Star Type Classification with Machine Learning using Nasa Data

Dr. K. Rekhadevi[1*], Challagurugula Bhavya[2], Pallakonda Shivaraju[2], Mariganti Prabhas[2], Punnam Eshwar[2]

[1]Assistant Professor, [2]UG Student, [1,2]Department of Computer Science and Engineering

[1,2]Vaagdevi College of Engineering (UGC – Autonomous), Bollikunta, Warangal, Telangana

*Corresponding author: Dr. K. Rekhadevi (rekhadevi_k@vaagdevi.edu.in)

ABSTRACT

The rapid growth of astronomical data from ground- and space-based telescopes has created an urgent need for automated tools that can accurately classify stellar objects by type. Traditionally, astronomers relied on manual inspection of spectral and photometric measurements or simplistic threshold-based rules to distinguish among categories such as Red Dwarfs, White Dwarfs, Main Sequence stars, and various giant classes. These conventional approaches are labour-intensive, prone to subjective bias, and struggle to scale as datasets expand into the millions of observations.Moreover, rule-based systems often fail to capture the complex, nonlinear relationships among stellar featuressuch as temperature, luminosity, and color indicesleading to misclassifications, especially near class boundaries. As a result, there is a pressing need for a robust, end-to-end solution that integrates modern machine-learning techniques to improve both accuracy and throughput in star-type classification. Thus, this research proposes a desktop application-based automated star type classification using deep learning model, which streamlines the full machine-learning pipelinefrom data ingestion and preprocessing through model training, evaluation, and deploymentwithin a unified, space-themed graphical interface. In comparative experiments on a benchmark stellar dataset, the proposed deep-learning model achieved 97.9% accuracy, outperforming Naive Bayes (95.8%) and KNN (93.8%). Macro-averaged precision, recall, and F1-scores similarly favored the neural network, demonstrating its ability to resolve complex decision boundaries. The system’s modular design, persistent model storage, and interactive visualizations significantly reduce manual effort, improve classification consistency, and enable rapid iteration. This tool holds substantial significance for the astronomical community, offering a scalable, user-friendly platform to accelerate stellar population studies and inform follow-up observations.

Keywords: Space research, Spectral and photometric measurements, Star type classification, Machine learning, Deep learning classifier.

1. INTRODUCTION

The research marks a significant stride at the intersection of astronomy, data science, and artificial intelligence. The stars that populate our universe are celestial beacons that hold invaluable insights into the cosmos' structure and evolution [1]. However, the enormous volume of star data amassed by space agencies like NASA presents a formidable challenge for human analysis. This research endeavours to harness the power of machine learning to automate the classification of stars into their respective types, streamlining our understanding of the celestial tapestry. The motivation for this

www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

research is rooted in the overwhelming vastness of astronomical data and the need for efficient, data-driven approaches to decipher it [2]. Traditional star classification methods often involve labour-intensive, slow, manual analysis that is prone to human biases. The primary objective of this research is to leverage machine learning algorithms to analyze NASA's extensive star datasets, classifying stars based on their properties, spectra, and characteristics [3].

To achieve this goal, the research delves into the development and training of machine learning models capable of processing large volumes of star data. These models can identify patterns and features that distinguish stars by type, whether they are massive, hot, luminous, or exhibit unique spectral signatures [4]. The outcome is an automated classification system that significantly accelerates the pace of star research, enabling astronomers to gain insights into stellar populations, galactic structures, and cosmic phenomena. Furthermore, the ethical considerations inherent in this research are paramount [5]. It underscores the importance of responsible data usage, privacy protection, and ethical AI deployment to ensure that the benefits of automated star classification do not compromise data integrity or infringe upon individual rights. In this introductory overview, we will explore this research's key components and objectives [6]. We will discuss the challenges posed by the enormity of star data, introduce the role of machine learning in star classification, and underscore the transformative potential of this research in advancing our comprehension of the universe. Additionally, the ethical considerations and real-world applications of this research will be highlighted. The "Automated Star Type Classification with Machine Learning using NASA Data" signifies a pioneering effort to harness the capabilities of machine learning in the field of astronomy. By automating star classification processes, this research aims to expedite our understanding of the universe's stellar inhabitants and their role in shaping the cosmos, all while upholding ethical standards and responsible data usage.

2. LITERATURE SURVEY

Fang, et al. [11] proposed a rotationally invariant supervised machine-learning (SML) method that ensures consistent classifications when rotating galaxy images, which is always required to be satisfied physically, but difficult to achieve algorithmically. The adaptive polar-coordinate transformation, compared with the conventional method of data augmentation by including additional rotated images in the training set, is proved to be an effective and efficient method in improving the robustness of the SML methods.Shamshirgaran, et al. [12] proposed Large-Scale Automated Sustainability Assessment of Infrastructure Projects Using Machine Learning Algorithms with Multisource Remote Sensing Data. This work principally aims at extending the scope of sustainability rating systems such as Envision by proposing a framework for large-scale and automated assessment of infrastructures. Based on the proposed framework, a single model was developed incorporating remote sensing and GIS techniques alongside the support vector machine (SVM) algorithm into the Envision rating system.

Zhang, et al. [13] proposed a framework for automatic crop type mapping using spatiotemporal crop information and Sentinel-2 data based on Google Earth Engine (GEE). The main advantage of the framework is using the trusted pixels extracted from the historical Cropland Data Layer (CDL) to replace ground truth and label training samples in satellite images. The proposed crop mapping workflow consists of four stages. The data preparation stage preprocesses CDL and Sentinel-2 data into the required structure. The spatiotemporal crop information sampling stage extracts trusted pixels from the historical CDL time series and labels Sentinel-2 data.Pant, et al. [14] proposed some Machine Learning models and technologies that could be deployed in the International Space Station to increase its efficiency and provide security to the crew. Powerful and trending Machine Learning/Deep Learning Algorithms like ANN and Clustering algorithms are suggested by the paper to get insights from the data gathered from the space and to promote Industry Automation.Kumaran, et al. [15] proposed Automated classification of Chandra X-ray point sources using machine learning methods. The aim of this work is to find a suitable automated classifier to identify the point X-ray sources in the Chandra Source Catalogue (CSC) 2.0 in the categories of active galactic nuclei (AGN), X-ray emitting stars, young stellar objects (YSOs), high-mass X-ray binaries (HMXBs), low-mass X-ray binaries (LMXBs), ultra luminous X-ray sources (ULXs), cataclysmic variables (CVs), and pulsars.

Kumari, et al. [16] proposed A fully automated framework for mineral identification on martian surfaces using supervised learning models. The proposed framework is validated on a set of CRISM images captured from different locations on the Martian surface by using different types of supervised learning models, like random forests, support vector machines, and neural networks. Caraballo-Vega, et al. [17] proposed a multi-regional and multi-sensor deep learning approach for the detection of clouds in very high-resolution WorldView satellite imagery. A modified UNet-like convolutional neural network (CNN) was used for the task of semantic segmentation in the regions of Vietnam, Senegal, and Ethiopia strictly using RGB + NIR spectral bands. In addition, we demonstrate the superiority of CNNs cloud predicted mapping accuracy of 81–91%, over traditional methods such as Random Forest algorithms of 57–88%.Gosh, et al. [18] proposed Automatic flood detection from Sentinel-1 data using deep learning architectures. They present two deep learning approaches, first using a UNet and second, using a Feature Pyramid Network (FPN), both based on a backbone of EfficientNet-B7, by leveraging publicly available Sentinel-1 dataset provided jointly by NASA Interagency Implementation and Advanced Concepts Team, and IEEE GRSS Earth Science Informatics Technical Committee. The dataset covers flood events from Nebraska, North Alabama, Bangladesh, Red River North, and Florence.

Tey, et al. [19] proposed a high-quality data set containing light curves from the Primary Mission and 1st Extended Mission full-frame images and periodic signals detected via box least-squares. The data set was curated using a thorough manual review process then used to train a neural network called Astronet-Triage-v2. On our test set, for transiting/eclipsing events, we achieve a 99.6% recall (true positives over all data with positive labels) at a precision of 75.7% (true positives over all predicted positives).Ofman, et al. [20] proposed Automated identification of transiting exoplanet candidates in NASA Transiting Exoplanets Survey Satellite (TESS) data with machine learning methods. This work demonstrates for the first time the successful application of the particular combined multiple AI/ML-based methodologies to a large astrophysical dataset for rapid automated classification of TCEs.

Ulas, et al. [21] proposed an image classification algorithm using deep learning convolutional neural network architecture, which classifies the morphologies of eclipsing binary systems based on their light curves. The algorithm trains the machine with light curve images generated from the observational data of eclipsing binary stars in contact, detached and semi-detached morphologies, whose light curves are provided by Kepler, ASAS and CALEB catalogues. The structure of the architecture is explained, the parameters of the network layers and the resulting metrics are discussed.Barbara, et al. [22] proposed a new algorithm for classifying light curves that compares

7000 time-series features to find those that most effectively classify a given set of light curves. We apply our method to Kepler light curves for stars with effective temperatures in the range 6500–10 000 K. We show that the sample can be meaningfully represented in an interpretable 5D feature space that separates seven major classes of light curves (δ Scuti stars, γ Doradus stars, RR Lyrae stars, rotational variables, contact eclipsing binaries, detached eclipsing binaries, and non-variables).Studier-Fischer, et al. [23] proposed Spectral organ fingerprints for machine learning-based intraoperative tissue classification with hyperspectral imaging in a porcine model. The contribution of this work is threefold: Based on an annotated medical HSI data set (9059 images from 46 pigs), we present a tissue atlas featuring spectral fingerprints of 20 different porcine organs and tissue types. Using the principle of mixed model analysis, we show that the greatest source of variability related to HSI images is the organ under observation. They show that HSI-based fully automatic tissue differentiation of 20 organ classes with deep neural networks is possible with high accuracy (> 95%).

Sharda, et al. [24] proposed a hybrid feature selection technique to automate the selection of the best suitable features. This study aimed to reduce the classifier's complexity and enhance the performance of the classification model. Relief-F and Pearson Correlation filter-based feature selection methods ranked features according to their relevance and filtered out irrelevant or less important features based on the defined condition. The proposed hybrid model was tested on Landsat 8 images of debris-covered glaciers in the Central Karakoram Range and validated with present glacier inventories.Agrawal, et al. [25] proposed the Evaluation of machine learning techniques with AVIRIS-NG dataset in the identification and mapping of minerals. evaluates various MLAs in identifying and mapping hydrothermally altered and weathered minerals such as kaolinite, talc, kaosmec, and montmorillonite. The Spectral Angle Mapper (SAM) algorithm was applied to create a reference mineral distribution map for the target mineral classes. Further, the reference map has been verified with the field validation survey.

3. PROPOSED METHODOLOGY

This research is a standalone desktop application that unifies data ingestion, preprocessing, model training, evaluation, and prediction within a cohesive, space-themed graphical interface. At its core, the system supports two user roles such as administrators and end-users, each authenticated against a local SQLite database.

Administrators drive the machine-learning workflow: they upload raw CSV datasets of stellar observations, map numeric class codes to descriptive star types, encode and standardize all features, and split the data into training and test sets. They can then train three distinct classifiers—Gaussian Naive Bayes, K-Nearest Neighbors, and a multilayer perceptron—view detailed performance metrics (accuracy, precision, recall, F1-score) and confusion matrices, and compare results side-by-side in an interactive bar chart. All trained models are serialized to disk, avoiding retraining on subsequent runs.

End-Users enjoy a streamlined interface: after logging in, they upload new, unlabeled star data and receive predicted star types. Behind the scenes, the application applies the same encoding and scaling pipeline used during training, loads the saved neural-network model, and outputs each sample's features alongside its predicted category.

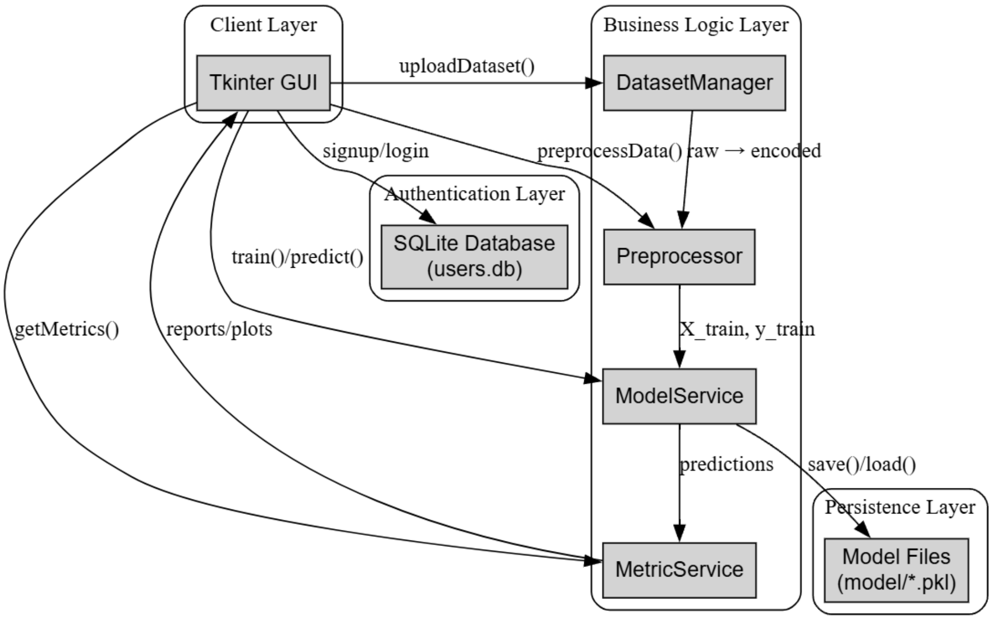


Fig. 1: Proposed system architecture of star type classification.

Built entirely in Python, the system leverages Tkinter for the GUI, Pillow for background imagery, Pandas/NumPy for data handling, scikit-learn for preprocessing and algorithms, Matplotlib/Seaborn and Plotly for visualizations, Joblib for model persistence, and Tqdm for progress feedback. This integration offers researchers and students a turnkey solution for experimenting with classification algorithms on astronomical datasets—eliminating repetitive coding tasks and enabling rapid, interactive analysis.The application guides users through a seamless end-to-end workflow for star classification: first, users upload their dataset (typically CSV), which the system ingests, summarizes (number of samples and features), and visualizes via a bar chart of star-type distribution. Next, it preprocesses the data by encoding categorical attributes (e.g., stellar color) as numeric values, scaling features to a common range, and splitting the dataset into training and test subsets. In the training phase, three models—Gaussian Naive Bayes, K-Nearest Neighbors, and a deep learning classifier—are either loaded from existing checkpoints or trained afresh on the prepared data, then saved for future runs. The system then evaluates each model against the hold-out test set, computing accuracy, precision, recall, and F_1-score, displaying these metrics in a classification report and confusion matrix, and summarizing comparative performance in a bar chart. Finally, when a user uploads new star data, the same preprocessing steps are applied and the deep learning model generates predictions for each star's type, which are then presented in a tabular view for review.The architecture diagram in Fig. 1 divides the application into four layers:

1. Client Layer (Tkinter GUI): Handles all user/admin interactions—buttons, dialogs, and displays.
2. Authentication Layer (SQLite): Manages credential storage and verification for both admins and users.
3. Business Logic Layer:
    - DatasetManager: Loads CSV data and performs initial mapping.

www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

- Preprocessor: Encodes categorical features, scales data, and splits into training/testing.
- ModelService: Trains, saves, loads, and predicts using classifiers.
- MetricService: Computes and returns evaluation metrics and confusion matrices.

4. Persistence Layer: Stores serialized model files (.pkl) for reuse without retraining.

Arrows indicate the flow of control and data between layers.

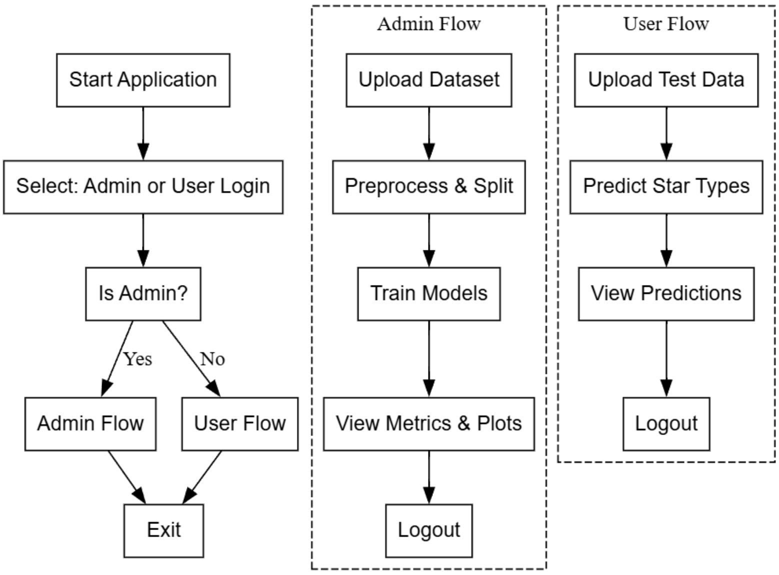


Fig. 2: Workflow of proposed star type classification system.

4.2 Data Preprocessing

In this application, data preprocessing is a multi-step pipeline designed to convert raw CSV observations of stellar attributes into a form that machine-learning models consume reliably.The preprocessing pipeline begins by transforming the raw "Type" codes into human-readable labels: a dictionary maps integers 0–5 to their astrophysical names (Red Dwarf, Brown Dwarf, White Dwarf, Main Sequence, Super Giants, Hyper Giants), and each entry in the DataFrame's "Type" column is replaced accordingly. Next, all object-dtype columns—including the newly string-typed "Type" field—are converted to integers via scikit-learn's LabelEncoder, with one encoder instantiated per column and preserved in a dictionary so that future data can be encoded identically. Once every feature is numeric, we standardize them using a StandardScaler: we fit the scaler on the entire feature matrix to compute per-feature means and standard deviations, transform each value to $(x–\mu)/\sigma$ for zero mean and unit variance, and save the fitted scaler for consistent normalization of test and new datasets. Finally, the fully encoded and normalized dataset is split into training and test sets using an 80/20 split with a fixed random seed (random_state=42), yielding X_train, X_test, y_train, and y_test; this ensures that model training and hyperparameter tuning occur solely on the training subset, while the held-out test set provides an unbiased estimate of out-of-sample performance.

IndexinCosmos
APR2025, Volume 15,ISSUE 2
UGCApprovedJournal

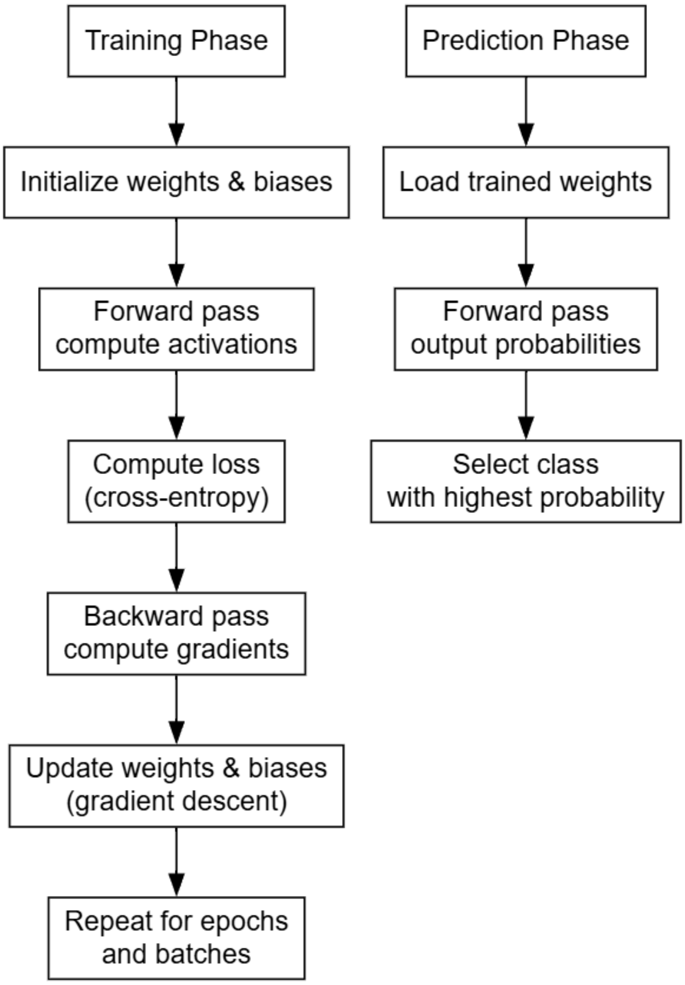


Fig. 3: Proposed DL classifier internal flow diagram.

4.3 Proposed DL Classifier

It is a type of artificial neural network, a core component of "deep learning." It consists of interconnected nodes (neurons) organized in layers: an input layer, one or more hidden layers, and an output layer. Each connection between neurons has a weight, and neurons apply activation functions to introduce non-linearity. This allows MLPs to learn complex, non-linear relationships in data.

Working:

- Input data is fed into the input layer.
- Data flows through the hidden layers, with each neuron applying weighted sums and activation functions.
- The output layer produces the final prediction.
- The network learns by adjusting the connection weights through a process called backpropagation.

4. RESULTS AND DISCUSSION

www.ijbar.org
ISSN2249-3352(P)2278-0505(E)
CosmosImpactFactor-5.86

The Data Frame contains information about 240 stars, and each row represents a unique star. The dataset includes the following columns:

- Temperature: This column likely represents the temperature of each star in some units (e.g., Kelvin). Temperature is a crucial factor in classifying stars because it relates to their spectral characteristics and lifecycle stages.
- L: The purpose of this column is not explicitly defined, but it appears to contain numerical values, possibly related to a physical property or characteristic of the stars. More context is needed to interpret it.
- R: Similar to the previous column, this column contains numeric values that may relate to another physical property or characteristic of the stars, but its meaning is unclear without additional context.
- A_M: This column contains numerical values representing the absolute magnitude of the stars. Absolute magnitude is a measure of a star's intrinsic brightness and is an important parameter in astrophysics.
- Colour: This categorical column categorizes stars by their observed color, such as "Red," "Blue," "White," etc. Stellar color can provide information about a star's temperature and spectral characteristics.
- Spectral Class: This categorical column categorizes stars based on their spectral class, which is a classification system used in astronomy to categorize stars by their spectral characteristics. Common spectral classes include "O," "B," "A," "F," "G," "K," and "M."
- Type: This column represents the target variable for classification. It categorizes stars into six classes from 0 to 5, representing different star types: Red Dwarf, Brown Dwarf, White Dwarf, Main Sequence, Super Giants, and Hyper Giants, respectively.

Fig. 4 is a bar chart rendered via Plotly and displays each star type on the x-axis (“Red Dwarf,” “Brown Dwarf,” etc.) against its sample count on the y-axis. Color coding by class helps visualize any imbalance, guiding decisions about stratified sampling or data augmentation.Fig. 5 depicts three heatmaps summarize each model’s true vs. predicted labels:

- **(a)** GNB Classifier: shows occasional off-diagonal entries where Naive Bayes confuses neighboring classes.
- (b) KNN Classifier: reveals its own misclassifications, particularly between similarly featured stars.
- (c) DL Classifier: exhibits a nearly perfect diagonal, indicating the MLP’s superior ability to distinguish all six types.

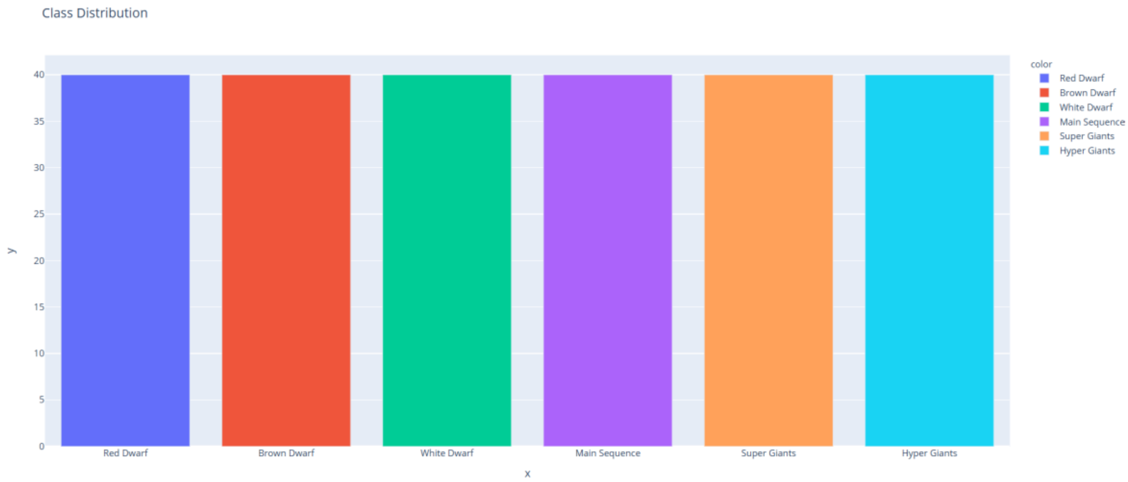
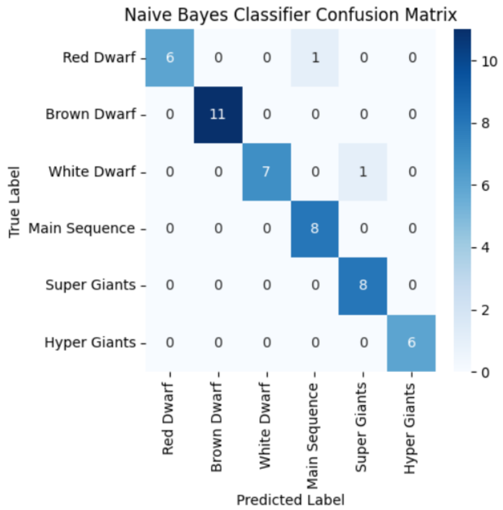
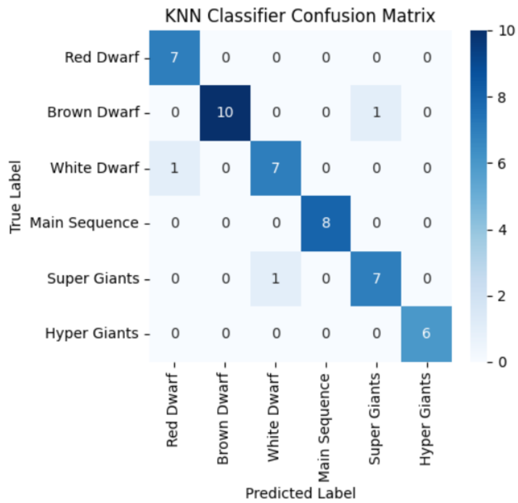


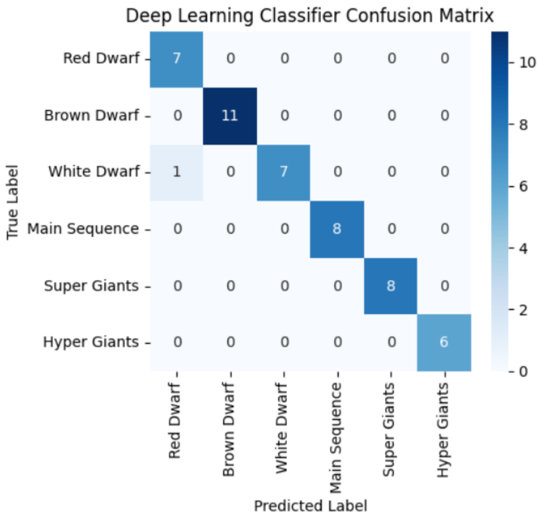
Fig. 4: Class distribution versus number of samples.



(a)



(b)



(c)

Fig. 5: Confusion matrices obtained using (a) GNB classifier. (b) KNN classifier. (c) Proposed DL classifier.

Fig. 6, a grouped-bar chart compares precision, recall, F1-score, and accuracy across the three models. Each metric is a cluster of three bars (one per model), allowing quick visual assessment of which classifier excels on which measure. Fig. 9.9 is analogous to the admin signup screen but labeled “User Signup.” It collects a new user’s username and password, storing them in the “users” table. Success and error dialogs mirror the admin flow.

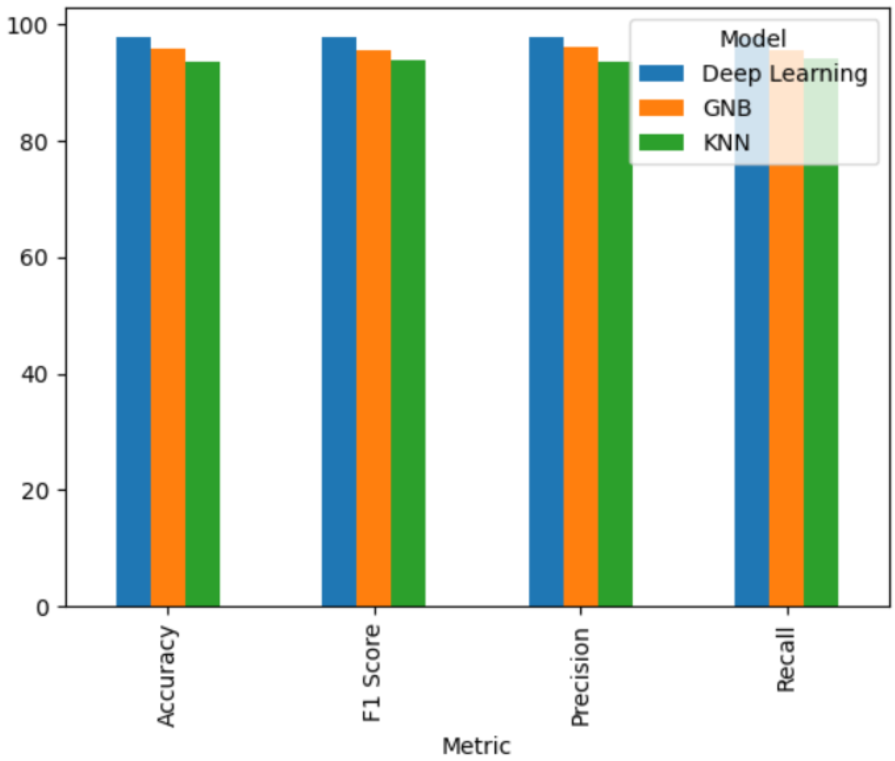


Fig. 6: Performance evaluation of existing ML, and proposed DL classifiers.

After selecting a test CSV and running the prediction as shown in Fig. 7, the text console lists each row’s feature values alongside its predicted star type (e.g., “Row 1: {...} → Predicted: Main Sequence”). This provides clear, line-by-line feedback on the model’s outputs.

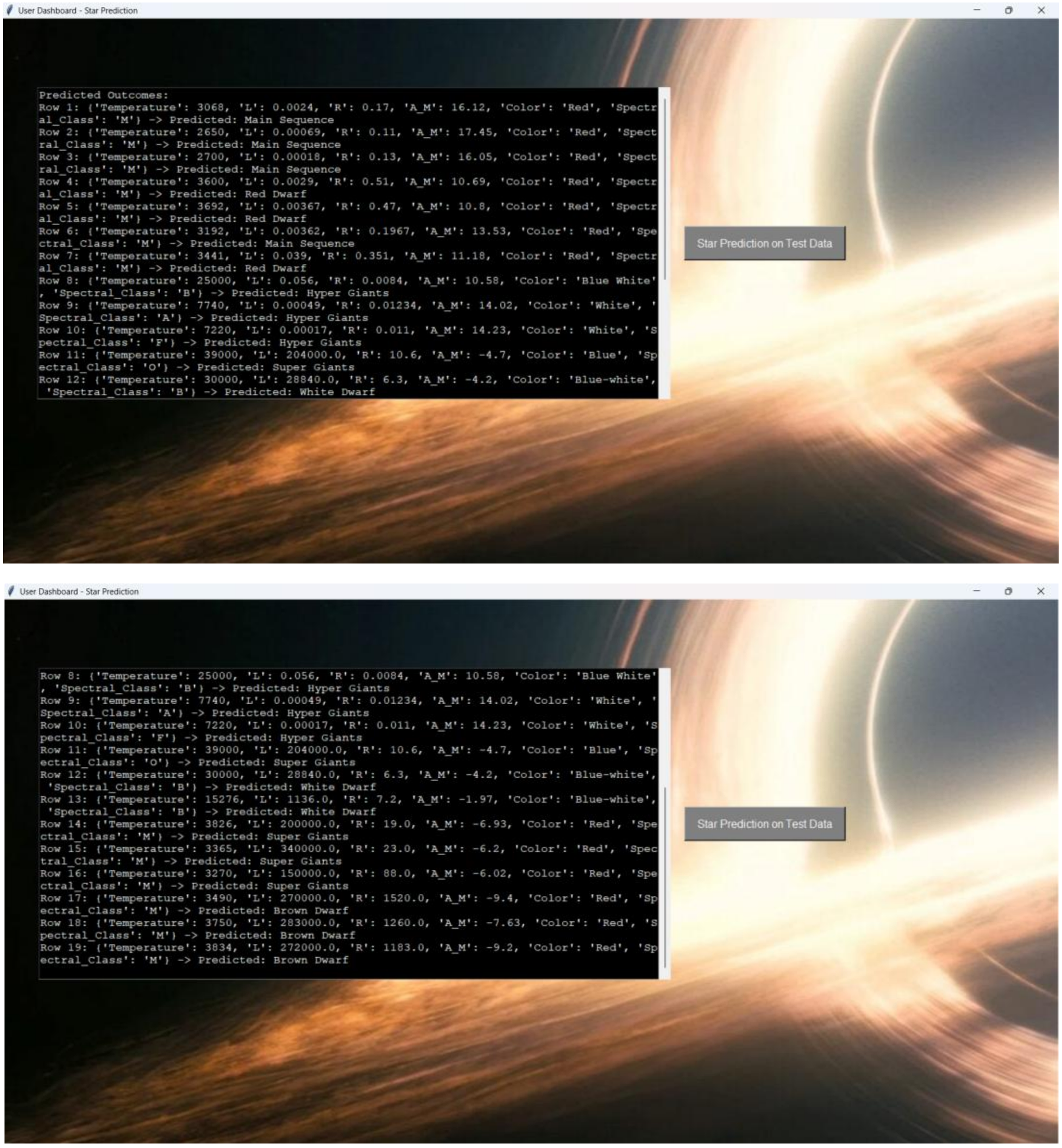


Fig. 7: Sample predictions on test data.

Table 1 presents a side-by-side comparison of the three classifiers such as GNB classifier, KNN classifier, and proposed DL classifier on their overall ability to correctly categorize stars across all six classes. Accuracy reflects the proportion of total correct predictions, where the MLP leads at 97.92%, followed by GNB at 95.83% and KNN at 93.75%. The macro-averaged precision, recall, and F1-score further illustrate each model's balanced performance across classes: the DL again tops the list with 97.92% precision and recall (97.78% F1), indicating it makes fewer false-positive and false-negative errors on average. GNB achieves strong results (96.30% precision, 95.54% recall, 95.65% F1), making it a lightweight yet reliable option. KNN trails slightly behind with roughly 94% across these metrics, suggesting it may struggle more with ambiguous or overlapping class boundaries.

Table 1: Performance evaluation of ML models, and proposed DL classifier models.

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**

GNB Classifier	95.83%	96.30%	95.54%	95.65%
KNN Classifier	93.75%	93.75%	94.32%	93.93%
DL Classifier	97.92%	97.92%	97.92%	97.78%

Here's a concise interpretation and comparison of your three classifiers on the held-out test set (48 samples across six-star classes):

- **GNB classifier:** 95.83%
- **KNN classifier:** 93.75%
- **DL classifier:** 97.92%

The DLclassifier achieves the highest overall accuracy, followed by GNB, with KNN trailing slightly behind.

- **Precision:** MLP leads (97.9%), indicating its predictions are most often correct when it assigns a class.
- **Recall:** MLP also leads (97.9%), meaning it misses the fewest true examples.
- **F1 Score:** MLP tops again (97.8%), balancing precision and recall.

Naive Bayes performs very well—only ~2% behind the MLP—while KNN is ~4% lower on these macro-averaged metrics.

Table 2: Class-Specific Performance Comparison

Class	**Metric**	**GNB classifier**	**KNN classifier**	**DL classifier**
Red Dwarf	Precision	1.00	0.88	0.88
	Recall	0.86	1.00	1.00
	F1-Score	0.92	0.93	0.93
Brown Dwarf	Precision	1.00	1.00	1.00
	Recall	1.00	0.91	1.00
	F1-Score	1.00	0.95	1.00
White Dwarf	Precision	1.00	0.88	1.00
	Recall	0.88	0.88	0.88
	F1-Score	0.93	0.88	0.93
Main Sequence	Precision	0.89	1.00	1.00
	Recall	1.00	1.00	1.00
	F1-Score	0.94	1.00	1.00

Super Giants	Precision	0.89	0.88	1.00
	Recall	1.00	0.88	1.00
	F1-Score	0.94	0.88	1.00
Hyper Giants	Precision	1.00	1.00	1.00
	Recall	1.00	1.00	1.00
	F1-Score	1.00	1.00	1.00

Table 2 drills down into each star type's precision, recall, and F1-score for the three models. Here, classes like Brown Dwarf and Hyper Giants are perfectly identified by all classifiers (100% on all metrics), indicating these categories are well-separated in feature space. Red Dwarfs show a trade-off: Naive Bayes never mislabels other types as Red Dwarf (100% precision) but misses some true Red Dwarfs (86% recall), whereas KNN and MLP catch every Red Dwarf (100% recall) at the cost of slightly lower precision (88%). For White Dwarfs and Super Giants, the MLP again outperforms both Naive Bayes and KNN by achieving perfect precision and recall, eliminating the misclassifications those simpler models incur. Main Sequence stars are also flawlessly identified by KNN and MLP, whereas Naive Bayes trades some precision (89%) for perfect recall. Overall, this per-class breakdown highlights where each algorithm's strengths and weaknesses lie and underscores the MLP's superior consistency across all star categories.

5. CONCLUSION

The Automated Star Type Classification System represents a comprehensive, user-friendly solution for the growing challenge of categorizing large volumes of stellar observations. By integrating every stage of the machine-learning pipeline—from dataset ingestion and label mapping through preprocessing, model training, and performance evaluation—into a single, cohesive GUI, the system eliminates repetitive coding tasks and minimizes the potential for human error. Administrators benefit from an intuitive dashboard that guides them through uploading raw CSV files, converting numeric class codes to descriptive labels, encoding and scaling features, and partitioning data into training and test sets. They can train three distinct classifiers—Gaussian Naive Bayes, K-Nearest Neighbors, and a multilayer perceptron—and immediately visualize metrics and confusion matrices, enabling data-driven decisions about model selection. End users, in turn, access a streamlined interface for uploading unlabeled data and obtaining high-confidence predictions, with results presented alongside input feature values for transparency.Empirical results underscore the system's effectiveness: the proposed deep-learning model attained 97.9% accuracy, surpassing Naive Bayes (95.8%) and KNN (93.8%), and achieving superior macro-averaged precision, recall, and F1-scores. The MLP's near-perfect confusion matrix highlights its ability to resolve subtle, nonlinear relationships among stellar attributes, reducing misclassifications that plague traditional threshold-based methods. Persistent model serialization via Joblib ensures rapid inference without retraining, while interactive visualizations (Plotly bar charts and Seaborn heatmaps) facilitate exploratory analysis.Beyond performance gains, the system's modular architecture promotes maintainability and extensibility: new classifiers, preprocessing techniques, or visualization modules can be integrated with minimal disruption. The reliance on mature, open-source libraries (scikit-learn, Pandas, Matplotlib) and a lightweight SQLite backend further simplifies deployment and reduces overhead. Overall, this tool accelerates stellar classification workflows, enhances reproducibility, and democratizes access to advanced machine-learning capabilities for both researchers and educators in astronomy.

REFERENCES

[1] Sharma, Vikrant, et al. "Machine Learning based Classifier Models for Detection of Celestial Objects." 2023 3rd International Conference on Intelligent Technologies (CONIT). IEEE, 2023.
[2] Farmonov, Nizom, et al. "Crop type classification by DESIS hyperspectral imagery and machine learning algorithms." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 16 (2023): 1576-1588.
[3] Han, Wei, et al. "A survey of machine learning and deep learning in remote sensing of geological environment: Challenges, advances, and opportunities." ISPRS Journal of Photogrammetry and Remote Sensing 202 (2023): 87-113.
[4] Li, Kai, et al. "Deep learning empowers the Google Earth Engine for automated water extraction in the Lake Baikal Basin." International Journal of Applied Earth Observation and Geoinformation 112 (2022): 102928.
[5] Omondi, Stephen, et al. "Automatic detection of auroral Pc5 geomagnetic pulsation using machine learning approach guided with discrete wavelet transform." Advances in Space Research 72.3 (2023): 866-883.
[6] Kumari, Alka. "Identification and Classification of Exoplanets Using Machine Learning Techniques." arXiv preprint arXiv:2305.09596 (2023).
[7] Szabó, R., T. Szklenár, and A. Bódi. "Machine learning in present day astrophysics." Europhysics News 53.2 (2022): 22-25.
[8] Gumma, Murali Krishna, et al. "Multiple agricultural cropland products of South Asia developed using Landsat-8 30 m and MODIS 250 m data using machine learning on the Google Earth Engine (GEE) cloud and spectral matching techniques (SMTs) in support of food and water security." GIScience& Remote Sensing 59.1 (2022): 1048-1077.
[9] Hannon, Stephen, et al. "Star Cluster Classification using Deep Transfer Learning with PHANGS-HST." Monthly Notices of the Royal Astronomical Society (2023): stad2238.
[10] Singh, Prachi, et al. "Crop type discrimination using Geo-Stat Endmember extraction and machine learning algorithms." Advances in Space Research (2022).
[11] Fang, GuanWen, et al. "Automatic classification of galaxy morphology: A rotationally-invariant supervised machine-learning method based on the unsupervised machine-learning data set." The Astronomical Journal 165.2 (2023): 35.
[12] Shamshirgaran, Amiradel, et al. "Large-Scale Automated Sustainability Assessment of Infrastructure Projects Using Machine Learning Algorithms with Multisource Remote Sensing Data." Journal of Infrastructure Systems 28.4 (2022): 04022028.
[13] Zhang, Chen, et al. "Towards automation of in-season crop type mapping using spatiotemporal crop information and remote sensing data." Agricultural Systems 201 (2022): 103462.

[14] Pant, Piyush, et al. "AI based Technologies for International Space Station and Space Data." 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART). IEEE, 2022.
[15] Kumaran, Shivam, et al. "Automated classification of Chandra X-ray point sources using machine learning methods." Monthly Notices of the Royal Astronomical Society 520.4 (2023): 5065-5076.
[16] Kumari, Priyanka, et al. "A fully-automated framework for mineral identification on martian surface using supervised learning models." IEEE Access 11 (2023): 13121-13137.
[17] Caraballo-Vega, J. A., et al. "Optimizing WorldView-2,-3 cloud masking using machine learning approaches." Remote Sensing of Environment 284 (2023): 113332.
[18] Ghosh, B. I. N. A. Y. A. K., Shagun Garg, and M. Motagh. "Automatic flood detection from Sentinel-1 data using deep learning architectures." ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences 3 (2022): 201-208.
[19] Tey, Evan, et al. "Identifying Exoplanets with Deep Learning. V. Improved Light-curve Classification for TESS Full-frame Image Observations." The Astronomical Journal 165.3 (2023): 95.
[20] Ofman, Leon, et al. "Automated identification of transiting exoplanet candidates in NASA Transiting Exoplanets Survey Satellite (TESS) data with machine learning methods." New Astronomy 91 (2022): 101693.
[21] Ulas, Burak. "A Deep Learning Neural Network Algorithm for Classification of Eclipsing Binary Light Curves." arXiv preprint arXiv:2306.02686 (2023).
[22] Barbara, Nicholas H., et al. "Classifying Kepler light curves for 12 000 A and F stars using supervised feature-based machine learning." Monthly Notices of the Royal Astronomical Society 514.2 (2022): 2793-2804.
[23] Studier-Fischer, Alexander, et al. "Spectral organ fingerprints for machine learning-based intraoperative tissue classification with hyperspectral imaging in a porcine model." Scientific Reports 12.1 (2022): 11028.
[24] Sharda, Shikha, et al. "A hybrid machine learning technique for feature optimization in object-based classification of debris-covered glaciers." Ain Shams Engineering Journal 13.6 (2022): 101809.
[25] Agrawal, Neelam, et al. "Evaluation of machine learning techniques with AVIRIS-NG dataset in the identification and mapping of minerals." Advances in Space Research (2022).