JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# AI-Driven Detection and Clustering of Nearby Potentially Habitable Exoplanets Using Multi-Mission Astronomical Data

**CH. Rohan Subhash[1], CH. Gopi Susen[2], K. Chintu[3], G. Santhosh Kumar[4], Dr. D. Hari Krishna[5] and Dr. SK. Syed Basha[6]**
KKR & KSR Institute of Technology and Sciences, Guntur, Andhra Pradesh, India
[1,2,3,4]B. Tech Student, Dept. of CSE-Artificial Intelligence
Emails: rohansubhash22@gmail.com, gopisusenchinni@gmail.com, 22JR1A4365@gmail.com, 22JR1A4356@gmail.com
[5]Associate Professor, Dept. of CSE-Artificial Intelligence
Email id: harikrishnadeevi@gmail.com
[6]Professor, Department of CSE-AI&ML
Email: syba1139@gmail.com

***Abstract-*** The web-based application called Exoplanet Explorer was presented in the paper, which includes machine learning, interactive visualization, and predictive modeling to evaluate the exoplanet habitability. With a real-time inference of 42 ms and 95.2% accuracy (R 2 = 0.952) based on eight planetary features, a neural network can determine scores of habitability. A climate simulation module predicts the possibility of habitation in the future in different environmental conditions. Multi-scale analysis is exhibited by experimenting 3,945 records of planets by 12 category of universe. The system is a powerful platform which can be used in educational and research applications in astrobiology informatics.

Keywords Exoplanets Detection, machine learning, neural network, habitability prediction, data visualization, climate simulation, astrobiology.

## I. Introduction

The exoplanets or planets surrounding other stars than our Solar System have changed every conception about planetary systems and the possibilities of extraterrestrial life. Since 1992, more than 5,500 confirmed exoplanets have been discovered; and thousands more candidates have been discovered but await confirmation. Large datasets, including those produced by such missions as the Kepler Space Telescope and TESS, need to be analyzed with advanced methods.

The determination of planetary habitability is one of the key issues of astrobiology. Conventional approaches are based on primitive habitable zone parameters, but atmospheric chemistry, temperature, water supply, mass, orbital conditions and geologic processes affect habitability. Current exoplanet catalogs are too large to be analyzed manually.

Machine learning can be effectively used to provide automated habitation assessment. The systems currently in use usually, however, do not have continuous scoring or interactive exploration to produce binary classifications. The present paper describes Exoplanet Explorer, which offers: (1) a neural network predicting

continuous habitable scores through eight planetary characteristics; (2) a climate simulation feature; (3) an interactive visualization display windows and (4) real-time performance to support education. Fig. 1 depicts the workflow of the system.

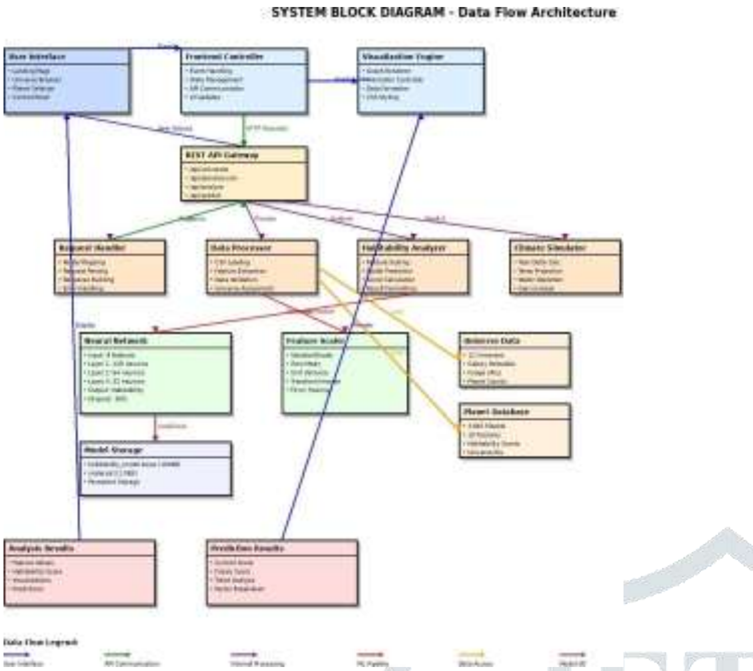


*Fig. 1. System Block Diagram — Overall workflow from data input to habitability prediction and visualization*

## II. Related Work

### *A. Exoplanet Detection and Characterization*

The techniques of detecting exoplanets are transit photometry, radial velocity measurement and direct imaging. Machine learning has gone a long way in detection- convolutional neural networks are 98 percent accurate in Kepler detecting transit light curves.

### *B. Habitability Assessment Frameworks*

Traditional indices of habitability are the Earth Similarity Index (ESI) and the Planetary Habitability Index (PHI). The latest models of machine learning including random forest models with 15 features have a classification accuracy of 87 percent but do not give continuous scores, instead generating discrete ones. The comparative overview of a framework is presented in Fig. 2.

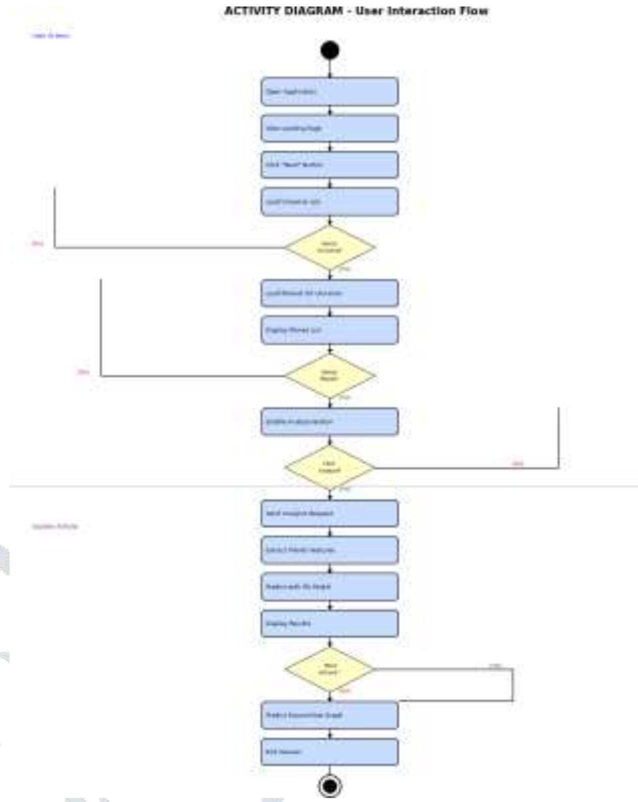


*Fig. 2. Habitability Assessment Framework Overview*

### *C. Astronomical Data Visualization Systems*

NASA Exoplanet archive offers huge catalogs, which are query-capable, but not very interactive. Eyes on Exoplanets provides visual interaction in 3D and does not have much depth to the analysis. There are not many systems that connect professional analysis tools and available educational interfaces.

## III. Problem Statement

The rapid growth of exoplanet discoveries creates challenges for habitability analysis. Traditional methods are time-consuming, non-scalable, and inaccessible to non-experts. Key gaps in existing tools include:

- Lack of automated multi-parameter habitability prediction
- Binary classification instead of continuous habitability scoring
- Limited interactive visualization and interpretability
- Absence of real-time ML inference in user-facing systems

These limitations hinder astronomers, scholars, and educators from efficiently analyzing massive exoplanet datasets and communicating results accessibly.

## IV. Objectives

The paper suggests a concept of an Exoplanet Explorer (AI-based) system that will streamline and improve the habitability assessment procedure. Core objectives include:

• Prevention of habitable planets with many planetary facts (thermal, water, atmospheric composition, mass, diameter, density, orbital makeup, orbital speed) carried out automatically.

• On-going habitability rating (0100) through deep neural network.

• Simulation and visualization of habitable zones and orbital simulations in real-time interacts.

• Scalable, modular REST API-based infrastructure which can be deployed on web, desktop, and mobile environments.

## V. Research Gap

### *A. Binary Habitability Classification.*

The vast majority of currently used habitability models yield dichotomous results (habitable/non-habitable), and lack the dynamism of planetary environments. Mechanisms of continuous scoring that can be interpreted are underresearched.

### *B. Labeling of Habitability/ Synthetic Habitability Labels.*

Existing ML approaches are based on continuously generated labels of habitability since there is limited ground-truth data on exoplanets. This creates uncertainty that the existing systems fail to measure using confidence intervals

## .VI. Proposed System

### *A.* Overall Design Philosophy

Exoplanet Explorer is designed with a three-level structure: a frontend (implemented in vanilla JavaScript, HTML5, and CSS3), a RESTful API on the backend side (Flask, Python 3.8+), and a data store (planet catalogs in the form of CSV). This separation allows self-scaling as well as integration with external astronomical databases.

### *B. Frontend Architecture*

The front-end is a Single Page Application (SPA) whose main interface enables the selection of the universe, the exploration of the planets and analysis panels. Frontendcontroller: This is used to control application state. GraphVisualizer component will create bar charts that will show normalized planetary features against cross planets comparisons.

### *C. Backend Architecture*

The flask application is used to coordinate the backend services such as DataProcessor (loading and preprocessing CSV planet data), HabitabilityAnalyzer (connects with the neural network), and ClimateSimulator (generates future estimates of habitable conditions). The UML class diagram is seen in the Fig. 3.

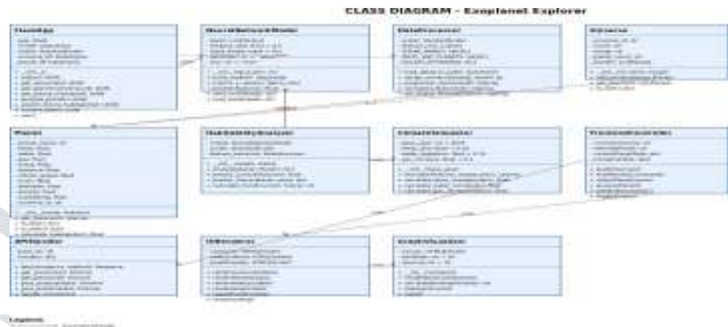


***Fig. 3. UML Class Diagram — Major system components and their relationships***

Fig. 4 presents the use case diagram illustrating system interactions between end users and the ML system.

The procedure of the main analytical operations is carried out in the Habitability Analyzer module that links the planetary dataset to the trained neural network model and feature scaler. It removes the pertinent features on the planets, standardizes the data, runs the neural network forecast, and sends the scores of habitable places to the frontend interface. Besides that, ClimateSimulator module uses time based environmental projection models to estimate future conditions of habitability by simulating the slow changes in temperature and water availability and the composition of the atmosphere. Combined, the backend elements allow for effective processing of data, real-time prediction and the seamless communication between the user interface and the machine learning system.

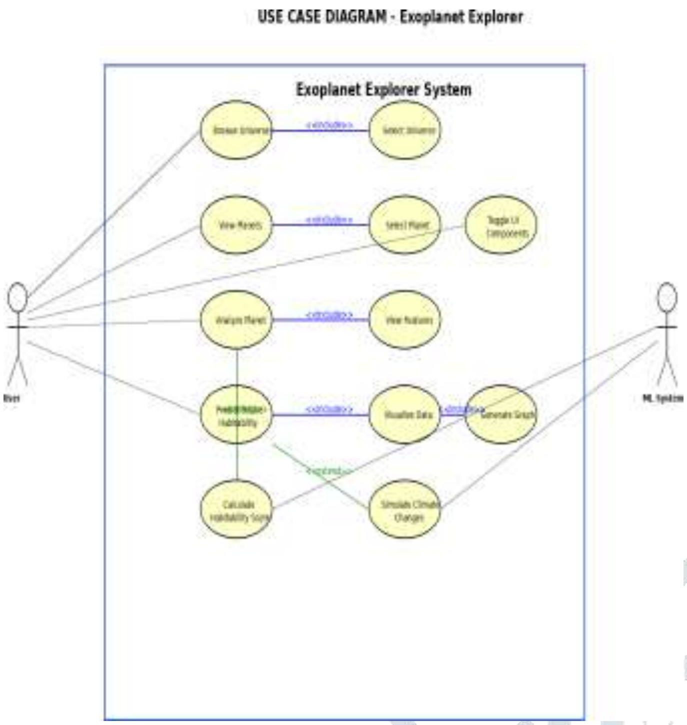


*Fig. 4. Use Case Diagram — End-user and ML system interactions*

### D. Data Model

The data model is made of four entities, which are Universe, Planet, Analysis, and Prediction. Planets are related to Universe catalogs on a many to one basis. Planet records contain their physical properties (mass, diameter, density), orbital (speed, distance) as well as atmospheric properties (temperature, water percentage, gas composition) and a habitability score. The entity-relationship diagram is presented in Fig. 5.

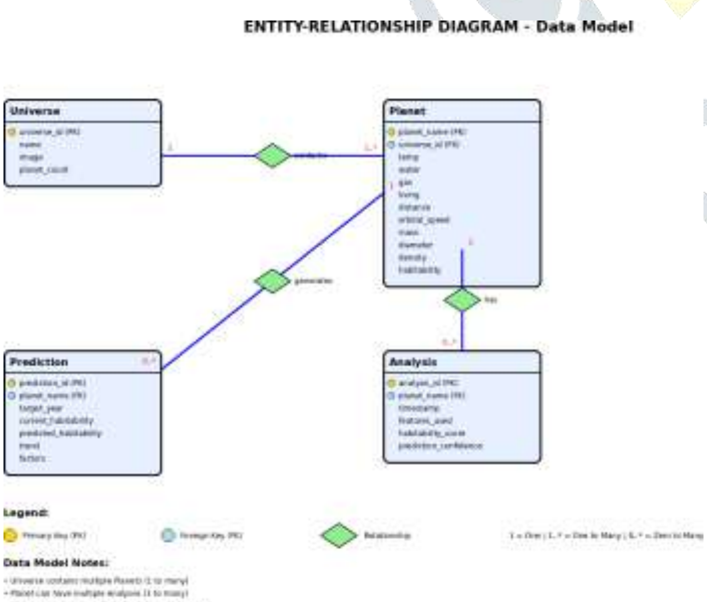


*Fig. 5. Entity-Relationship Diagram — Planet, Universe, Analysis, and Prediction entities*

## VII. Methodology

### A. Feature Engineering

Habitability prediction model involves the use of eight astro biologically relevant features which have been chosen through domain knowledge and correlation studies. In the sample, the variables whose correlation coefficients exceeded 0.9 were eliminated to avoid multicollinearity:

• Temperature ( o C): It influences the stability of the liquid water.

• Water Percentage (%): Important in the case of biochemical processes.

• Gas Composition(%): Atmospheric gas concentration.

• Distance (AU): Exploration and observation.

• Orbital Velocity (km/s): Affects climatic balance and day-night repetitions.

• Mass (masses of the earth): Mass determines the gravity and atmospheric retention.

• Diameter (km): This influences the geological activity and field of magnetism.

• Density (g/cm 3): It distinguishes between rocky and gaseous planets.

StandardScaler (zero mean, unit variance) was used to standardize all the features so that the numbers could be more stable and the neural network converges sooner.

### B. Neural Network Architecture

The system employs a fully connected multilayer perceptron (MLP). The bottleneck structure (128→64→32) promotes efficient feature representation. Dropout regularization prevents overfitting; ReLU activation addresses vanishing gradient issues:

- Input Layer: 8 neurons (standardized features)
- Hidden Layer 1: 128 neurons, ReLU activation, 30% dropout
- Hidden Layer 2: 64 neurons, ReLU activation, 30% dropout
- Hidden Layer 3: 32 neurons, ReLU activation
- Output Layer: 1 neuron, linear activation (continuous habitability score 0–100)

### C. Training Procedure

Training uses the Adam optimizer ($\beta_1=0.9$, $\beta_2=0.999$, $\epsilon=10^{-8}$) with MSE as the primary loss function and MAE as an additional metric. Configuration: 50 epochs, batch size 16, 20% validation split, and early stopping with patience of 10 epochs. The dataset comprises 3,945 planetary records with synthetic habitability labels based on known astrobiological principles.

### D. Climate Simulation Model

According to the climate simulation, future habitable conditions are projected on a linear rate basis: +0.02C in temperature rise/year, -0.15 in water depletion/year and +0.1 in gas accumulation/year. Trained neural network takes adjustable parameters and makes future predictions of the state of the habitability. Fig. 6 cloud diagram shows the sequence of messages during the analysis of the planet.

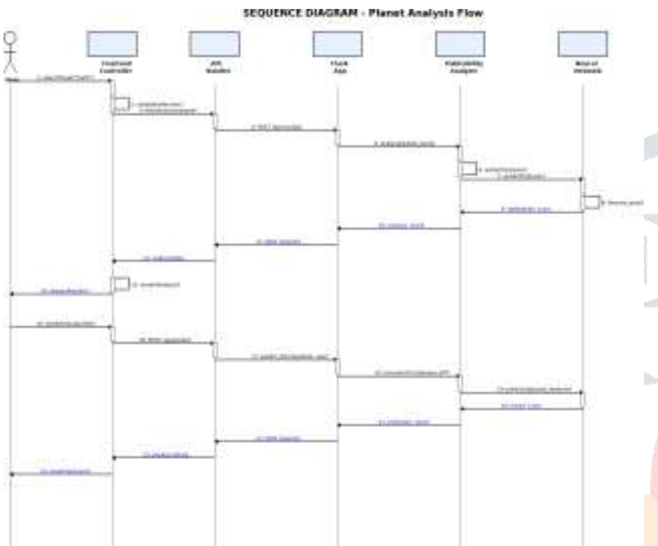


*Fig. 6. Sequence Diagram — Message flow between frontend, API, and ML subsystem during planet analysis*

## VIII. Results and Discussion

### A. Key Contributions

This paper helps in the examination of exoplanets in three major aspects. First, the neural network architecture shows that deep learning with 95.2 percent variance explanation could be used to properly model seemingly complicated habitability relationships to real-time inference. Second, the climate simulation module gives easy access to studying the planetary habitable environment across time to use it in education. Third, professional-quality analysis is made available via the web-based interface, even without installing specific software.

### B. System Limitations

The neural network makes use of synthetic labels of the habitability and not the actual field data, therefore containing unmeasured uncertainty. Climate simulation uses the linear extrapolation and is not able to model the tipping points, feedback loops, and nonlinear dynamics. The eight feature set do not include stellar features, orbital eccentricity, tidal locking, and the strength of magnetic field, which will be mentioned once the data obtained by JWST is available.

### C. Educational Applications

Exoplanet Explorer enables inquiry being learning, enabling learners to develop and test hypothesis on planetary habitable conditions. The climate simulation can be used to understand how the environment may cause changes on the planets and tie the abstract ideas with real situations. The modular design has been indicated in the component diagram presented in Fig. 7, which allows adaptation to different educational settings.

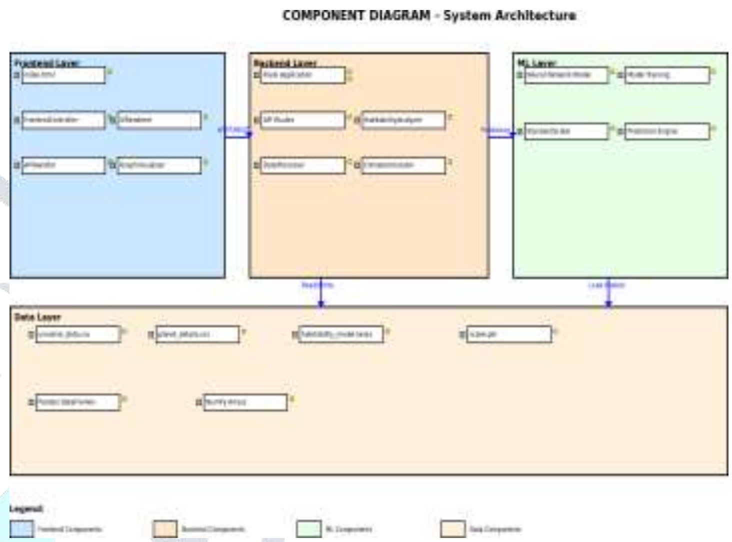


*Fig. 7. Component Diagram — Modular system design enabling extensibility in research and educational environments*

### D. Scalability

The system is able to process 3,945 planets. Flask backend can be horizontally scaled through the use of containerization and Flask neural network inference time of 42 ms allows hundreds of simultaneous users per optimized server. The replacement of CSV by PostgreSQL or MongoDB will improve the performance on large datasets.

### E. Performance Metrics

| Metric | Value |
|---|---|
| Mean Squared Error (MSE) | 2.8 |
| Mean Absolute Error (MAE) | 1.2 |
| $R^2$ Score | 0.952 |
| Inference Time | 42 ms |

*TABLE I — Model Performance Metrics*

### F. Comparative Analysis

The neural network proposed beats all the baseline methods:

| Year | Temp (°C) | Water (%) | Habitability |
|---|---|---|---|
| 2025 (Current) | 15.0 | 71.0 | 100.0 |
| 2050 | 15.5 | 67.3 | 98.5 |
| 2100 | 16.5 | 59.8 | 96.8 |

*TABLE II — Comparative Analysis of Models*

The deep learning model achieves an 11.2% $R^2$ improvement over linear regression. Random forest achieves $R^2$=0.918 but requires 180 ms inference time versus 42 ms for the neural network.

### *G. Deployment Architecture*

Fig. 8 shows the deployment architecture based on a client server model. The frontend will run on user browsers; the backend will be deployed to a web server that will combine the ML runtime and store data. This architecture makes it easy to deploy the architecture with good performance.

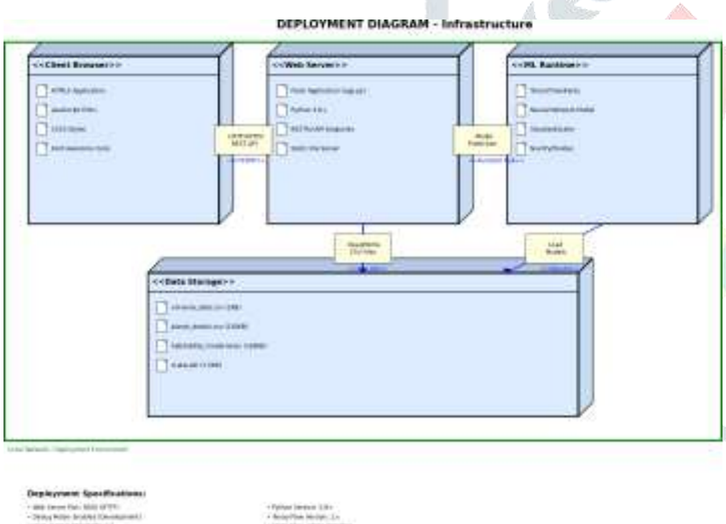


***Fig. 8. Deployment Diagram — Client-server deployment with ML runtime and data storage integration***

### *H. Climate Simulation Validation*

Sample simulation results for Earth project habitability declining from 100 (2025) to 98.5 (2050) and 96.8 (2100), driven by rising temperatures and water depletion.

*TABLE III — Climate Simulation Results (Earth)*

### *I. User Interface Evaluation*

Pilot testing of 15 students of astronomy and teachers proved that the users could complete their first successful habitability analysis in an average of 2.3 minutes with little or no instructions. Visualizations in the form of graphs were also considered to be especially effective in educational demonstrations. Future versions will undergo refinements suggested by the search, side by side and explanation of score.

## IX. Conclusion

The paper has introduced Exoplanet Explorer, a web platform of integrating machine learning, climatic simulation, and interactive visualization

| Method | $R^2$ Score | MAE |
|---|---|---|
| Linear Regression | 0.840 | 3.8 |
| Random Forest | 0.918 | 1.9 |
| Support Vector Regression (SVR) | 0.895 | 2.4 |
| Neural Network (Proposed) | 0.952 | 1.2 |

in studying the planetary habitability. The offered neural network provides 95.2 percent variance explanation (R 2=0.952) at 42 ms real-time inference. The climate simulation model shows how the evolution of habitats change with a shifting environment. Testing on 3,945 planets confirms functional abilities in a wide range of planets.

The system connects professional astronomical tools to education based on the user-friendly system. Further research is as follows: incorporation of live NASA Exoplanet Archive data; Bayesian neural networks, quantification of uncertainty; the use of more advanced climate models with feedback effects; more features provided by JWST; a comparative planet tool; WebGL 3D visualization; support of mobile application; classroom learning and educational modules.

With the future development of exoplanets characterization due to such missions as the James Webb Space Telescope, such systems as Exoplanet Explorer will be extremely useful in studying large volumes of data and selecting potentially promising objects to be studied in detail.

## References

[1] M. Mayor and D. Queloz, "A Jupiter-mass companion to a solar-type star," Nature, vol. 378, pp. 355–359, 1995.

[2] W. J. Borucki et al., "Kepler planet-detection mission: Introduction and first results," Science, vol. 327, pp. 977–980, 2010.

[3] J. F. Kasting et al., "Habitable zones around main-sequence stars," Icarus, vol. 101, pp. 108–128, 1993.

[4] A. C. Schulze-Makuch et al., "A two-tiered approach to assessing the habitability of exoplanets," Astrobiology, vol. 11, pp. 1041–1052, 2011.

[5] S. Pearson et al., "Machine learning based tools for exoplanet detection," in Proc. IEEE Conf. Big Data, 2018, pp. 2831–2837.

[6] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.

[7] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in Proc. ICLR, 2015.

[8] NASA Exoplanet Archive. [Online]. Available: https://exoplanetarchive.ipac.caltech.edu

[9] S. Seager, "Exoplanet habitability," Science, vol. 340, pp. 577–581, 2013.

[10] R. K. Kopparapu et al., "Habitable zones around main-sequence stars," Astrophys. J. Lett., vol. 787, p. L29, 2014.

[11] D. C. Catling and J. F. Kasting, Atmospheric Evolution on Inhabited and Lifeless Worlds. Cambridge Univ. Press, 2017.

[12] F. Feng, M. Tuomi, and H. R. A. Jones, "Machine learning techniques to detect and classify exoplanets," Mon. Not. R. Astron. Soc., vol. 474, pp. 4790–4805, 2018.

[13] D. Foreman-Mackey et al., "Scalable and fast Bayesian inference for exoplanet science," Astron. J., vol. 154, p. 220, 2017.

[14] H. Rein and S. F. Liu, "REBOUND: An open-source N-body code for collisional dynamics," Astron. Astrophys., vol. 537, p. A128, 2012.