Hybrid Attention-DenseNet for Robust Exoplanet Validation

Sneha Basker
CentraleSupélec
sneha.basker@student-cs.fr

Julian Dequatre
CentraleSupélec
julian.dequatre@student-cs.fr

Abstract

This report details the development of a Machine Learning pipeline designed to identify exoplanets using photometric data from the NASA Kepler mission. The project addresses the problem of distinguishing planetary candidates from astrophysical false positives.

Three distinct approaches were evaluated. First, a Feed-Forward Neural Network (ExoNet) achieved 86% accuracy but suffered from a high false positive rate. Second, a Convolutional Neural Network (CNN) applied to raw light curves failed to generalize due to data scarcity. Finally, a ***Hybrid Architecture*** *combining Multi-Head Self-Attention and DenseNet blocks was developed. This final model successfully solved the class imbalance issue, achieving a global accuracy of 91.5% and an ROC-AUC of 0.979.*

1. Introduction and Context

The search for exoplanets is a data-intensive field in modern astrophysics. The NASA Kepler space telescope detected planets using the *transit method* [2], which involves monitoring stars for periodic dips in brightness caused by a planet crossing the star's line of sight.

However, not all detected dips are planets. Many are **False Positives** caused by background noise, instrument errors, or binary star systems. Validating these signals manually is time-consuming. This project aims to automate this validation process using a Deep Learning approach.

We benchmarked three architectures: a standard MLP, a 1D-CNN on raw time-series data, and a novel Hybrid model utilizing attention mechanisms.

2. Problem Statement

The objective is to solve a **Supervised Binary Classification** problem.

- **Input (X):** A set of physical characteristics describing the signal (KOI - Kepler Object of Interest) and the host star.
- **Target (y):** A binary label indicating if the object is a *Planet* (1) or a *False Positive* (0).

The core challenge lies in learning the non-linear relationships between stellar properties and transit signals to accurately classify ambiguous candidates.

3. Methodology (Phase 1: MLP Baseline)

3.1. Data Acquisition and Engineering

The dataset was constructed by merging two primary sources:

1. **KOI Data:** Contains transit parameters (Period, Depth, Duration).
2. **Stellar Data:** Contains host star properties (Temperature, Gravity, Radius).

The merge was performed on the unique identifier `kepid`.

3.1.1. Target Definition

The original dataset contained three labels: `CONFIRMED`, `CANDIDATE`, and `FALSE POSITIVE`. These were binarized as follows:

- **Class 1 (Planet):** Union of CONFIRMED and CANDIDATE.
- **Class 0 (False Positive):** FALSE POSITIVE.

3.1.2. Feature Selection

Ten features were initially selected to train the baseline model, including:

- *Orbital:* Period, Time of Transit, Duration.
- *Signal:* Transit Depth, Planetary Radius (R_p), Insolation Flux.
- *Stellar:* Effective Temperature (T_{eff}), Surface Gravity ($\log g$), Stellar Radius.

3.2. Preprocessing Pipeline

Before training, the data underwent rigorous preprocessing:

1. **Imputation:** Missing values (NaNs) were filled using the **median** strategy via `SimpleImputer`.
2. **Scaling:** All features were normalized using `StandardScaler` ($mean = 0, variance = 1$) to facilitate gradient descent convergence.

Figure 1. Model Training Loss Curve (MLP)

3. **Splitting:** The data was split into Training (80%), Validation (10%), and Test (10%) sets using stratified sampling.

3.3. Model Architecture: ExoNet

The baseline model is a Multi-Layer Perceptron (MLP) implemented in **PyTorch** [8]. The architecture is defined as follows:

- **Input Layer:** 10 neurons.
- **Hidden Layer 1:** 64 neurons $\rightarrow$ Batch Normalization $\rightarrow$ ReLU Activation $\rightarrow$ Dropout ($p = 0.3$).
- **Hidden Layer 2:** 32 neurons $\rightarrow$ Batch Normalization $\rightarrow$ ReLU Activation.
- **Output Layer:** 1 neuron (Logit).

3.4. Training Strategy

- **Hardware:** Training was accelerated using a GPU (CUDA/RTX 2000 Ada).
- **Loss Function:** `BCEWithLogitsLoss`.
- **Optimizer:** Adam with a learning rate of 3×10^{-4}.
- **Duration:** 100 Epochs.

4. Results and Analysis (MLP)

4.1. Performance Metrics

The model was evaluated on a test set containing 1642 samples. The global accuracy reached **0.86**. The interpretation is as follows:

- **Planets (Class 1):** Recall of 0.96.
- **False Positives (Class 0):** Recall is only 0.47.
- **Precision Gap:** The precision for False Positives (0.77) is lower than for Planets (0.87).

5. Phase 2: Investigation of Raw Light Curves (CNN)

To mitigate the feature limitations, we explored a Deep Learning approach using raw photometric time-series data using a 1D Convolutional Neural Network (CNN).

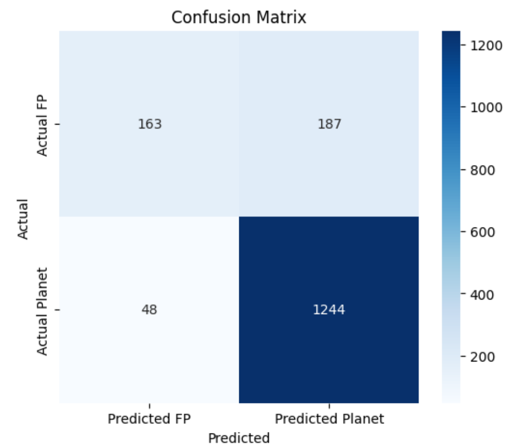

Figure 2. Confusion Matrix for the MLP Model

5.1. Methodology

We utilized the `lightkurve` library [5] to retrieve light curves. The data was normalized, phase-folded using the orbital period, and interpolated to a fixed grid of 1024 bins.

5.2. Results and Failure Analysis

Despite robust architecture choices, the CNN approach significantly underperformed. The evaluation on the test set ($N = 658$) yielded an accuracy of only **47%**. The model effectively collapsed, failing to learn distinctive features from the noisy flux data given the small dataset size (~2,600 training samples).

6. Phase 3: The Hybrid Attention-DenseNet Solution

Given the limitations of the simple MLP and the failure of the CNN, we developed a third, highly optimized architecture. This **Hybrid Model** applies modern Deep Learning techniques (Attention, DenseNets) to an expanded set of tabular features.

6.1. Advanced Methodology

6.1.1. Data Re-acquisition and Feature Engineering

Unlike the first phase which relied on a manual merge of stellar and KOI tables, we switched to the official **Kepler Cumulative Data** from the NASA Exoplanet Archive [1]. This comprehensive dataset allowed us to access a wider range of derived parameters.

- **Feature Expansion:** We increased the input dimensionality from 10 to **20 features** based on a correlation analysis. Crucially, this new dataset includes the *False Positive Flags* (e.g., `koi_fpflag_ss`, `koi_fpflag_co`), which proved highly correlated with the target but were absent in the initial baseline.

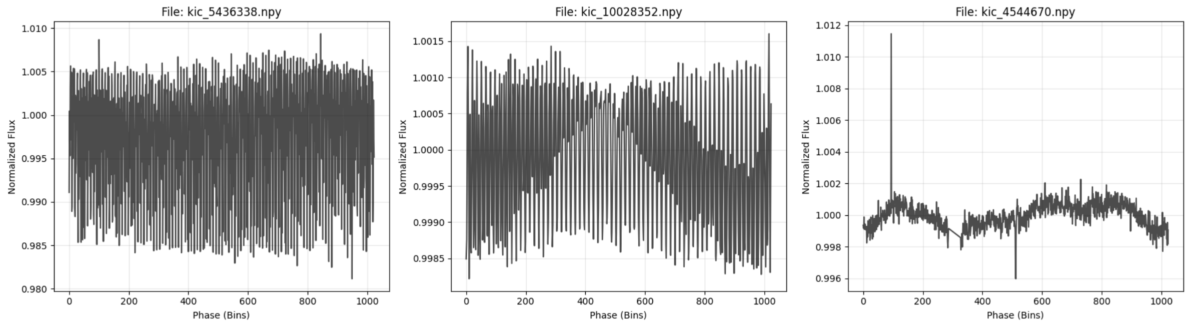

Figure 3. Sample Raw Light Curves processed for the CNN input.

- **Log-Transformation:** A `log1p` transformation was applied to skewed features (e.g., `koi_period`, `koi_depth`) to reduce the impact of heavy tails.
- **Robust Scaling:** We replaced StandardScaler with `RobustScaler`, which utilizes the Interquartile Range (IQR) to minimize the influence of outliers typical in astronomical data.

6.1.2. Architecture: Attention + DenseNet

The model architecture is designed to capture complex feature interactions:

1. **Multi-Head Self-Attention:** The input is projected to 128 dimensions and passed through a 4-head attention block [10]. This allows the model to weigh the importance of specific features relative to others (contextual understanding).
2. **Dense Block:** Inspired by DenseNet [4], this block contains 3 layers with a growth rate of 32. Each layer receives inputs from all previous layers, improving gradient flow.
3. **Squeeze-Excitation (SE):** A final residual block includes an SE module [3], which adaptively recalibrates channel-wise feature responses to emphasize informative features and suppress noise.

6.2. Training Strategy

To prevent the stagnation observed in the MLP, we employed:

- **Mixup ($\alpha = 0.2$):** A data augmentation technique [11] that trains the model on linear combinations of examples.
- **Label Smoothing (0.1):** Prevents the model from becoming over-confident [9].
- **Optimization:** AdamW optimizer [7] with Cosine Annealing Warm Restarts ($T_0 = 10$) [6].

6.3. Final Results

The Hybrid model was evaluated using Stratified 5-Fold Cross-Validation and a final hold-out test set. It significantly outperformed all previous approaches.

- **Global Accuracy: 91.54%** (vs 86% for MLP).

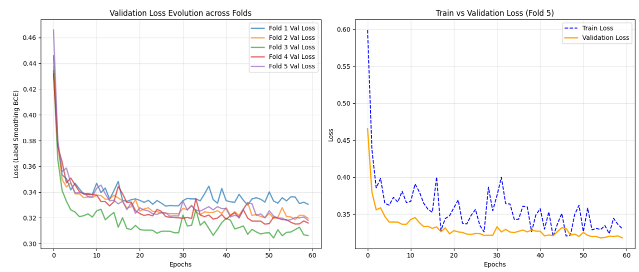

Figure 4. Hybrid Model Loss. Note the stable convergence of Train vs Validation loss (right), unlike the MLP.

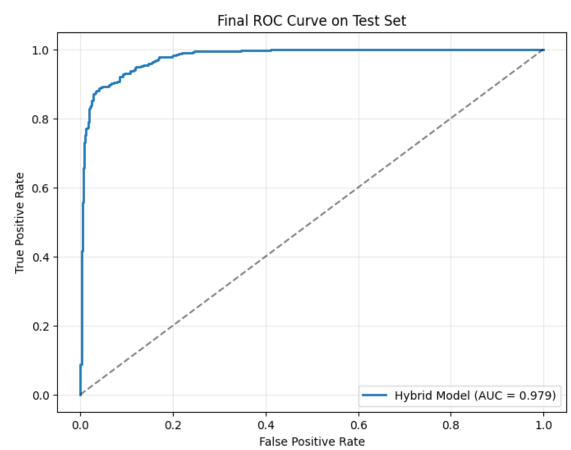

Figure 5. Final ROC Curve for the Hybrid Model (AUC = 0.979)

- **ROC AUC: 0.9786**, indicating excellent class separability.
- **False Positive Reduction:** Crucially, the model reduced Type I errors (False Positives) to just **30** samples, achieving a Precision of **93.36%**.

The Confusion Matrix below highlights the drastic reduction in False Positives compared to the initial MLP (which had ∼160 errors).

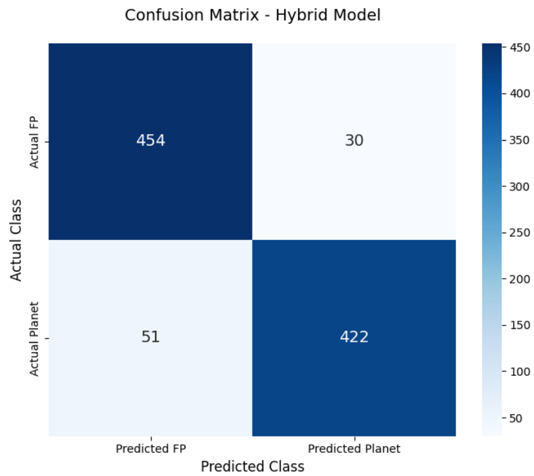

Figure 6. Confusion Matrix for the Hybrid Model. Note the low False Positive count (30).

7. Final Conclusion

This project demonstrates the evolution of a detection pipeline from a basic neural network to a state-of-the-art hybrid architecture. While the initial **ExoNet (MLP)** suffered from bias towards the majority class and the **CNN** failed due to data scarcity, the final **Hybrid Model** successfully bridged the gap.

By combining rigorous feature engineering (Log-transform, RobustScaler) with advanced architectural components (Self-Attention, Squeeze-Excitation) and regularization (Mixup), we achieved a robust classifier with **91.5% accuracy** and high reliability in filtering False Positives. This model represents a viable tool for automating the validation of Kepler candidates.

References

[1] Nasa exoplanet archive. https://exoplanetarchive.ipac.caltech.edu/, 2025. Accessed: 2025-12-10. 2

[2] William J Borucki et al. The kepler mission. *Science*, 327 (5968):977–980, 2010. 1

[3] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7132–7141, 2018. 3

[4] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4700–4708, 2017. 3

[5] Lightkurve Collaboration et al. Lightkurve: Kepler and TESS time series analysis in Python. *Astrophysics Source Code Library*, 2018. 2

[6] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 3

[7] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 3

[8] Adam Paszke et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 2

[9] Christian Szegedy et al. Rethinking the inception architecture for computer vision. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 3

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 3

[11] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 3