POOJITHA K

au211521104105

CELESTIAL OBJECTS CLASSIFICATION(CNN)

CELESTIAL BODIES CLASSIFICATION

A CONVOLUTIONAL NEURAL NETWORK(CNN)-based Approach

AGENDA

- Problem Statement
- Proposed system/solution
- Who are End Users
- System development approach
- Algorithm and deployment
- Model Overview
- Result
- Conclusion
- References

PROBLEM STATEMENT

The classification of celestial objects, such as stars, galaxies, poses a significant challenge in astronomy due to the vast amounts of observational data collected by modern telescopes. Manual classification methods are time-consuming and impractical for large datasets, while traditional automated techniques often lack accuracy or require complex algorithms. Therefore, there is a pressing need for efficient and accurate automated classification systems to categorize celestial objects based on their characteristics.

PROPOSED SYSTEM/SOLUTION

We propose to develop a Convolutional Neural Network (CNN)-based classification system for distinguishing between stars and galaxies in celestial images. The CNN will be trained on a dataset containing labeled images of stars and galaxies. By learning the distinguishing features from these images, the CNN will be able to accurately classify new images of celestial objects.. Here's how our proposed system works:

1. Data Collection and Preparation:

- Acquire a comprehensive dataset containing labeled images of stars and galaxies. The dataset should cover a wide range of celestial objects with varying characteristics.
- Preprocess the images by resizing them to a standard size, normalizing pixel values, and applying augmentation techniques to enhance the diversity of the dataset.

2. Model Development:

- Design and train a CNN architecture tailored for celestial object classification. Experiment with different network architectures, including the number of convolutional layers, filter sizes, and activation functions, to optimize performance.

- Utilize transfer learning techniques by fine-tuning pre-trained CNN models (e.g., VGG, ResNet) on astronomical datasets to leverage their feature extraction capabilities and improve classification accuracy.

3. Training and Validation:

- Split the dataset into training, validation, and test sets to facilitate model training and evaluation.

- Train the CNN model using the training set and validate its performance using the validation set. Monitor key metrics such as loss and accuracy during training to assess model convergence and performance.

4. Hyperparameter Tuning and Optimization:

- Fine-tune model hyperparameters such as learning rate, batch size, and dropout rate to optimize model performance and prevent overfitting.

- Implement regularization techniques such as dropout and L2 regularization to enhance model generalization and reduce the risk of overfitting.

5.Evaluation and Validation:

- Evaluate the trained CNN model using the test set to assess its classification accuracy, precision, recall, and F1-score.

- Conduct extensive performance analysis, including confusion matrix visualization and ROC curve analysis, to gain insights into the model's strengths and limitations.

6.Deployment and Integration:

- Deploy the trained CNN model as a standalone application, web service, or API to provide users with access to celestial object classification functionality.

- Integrate the classification system into existing astronomical data processing pipelines and software platforms to automate the classification of large-scale astronomical datasets.

7.User Interface and Interaction:

- Develop a user-friendly interface for interacting with the classification system, allowing users to upload images, visualize classification results, and access additional information about classified celestial objects.

- Provide features for batch processing and analysis, enabling users to classify multiple images simultaneously and perform advanced data exploration tasks.

8.Maintenance and Updates:

- Establish a system for monitoring and maintaining the classification system, including regular updates to accommodate new data, improve model performance, and address emerging challenges in celestial object classification.

- Incorporate feedback from users and stakeholders to continuously refine and enhance the system's functionality, usability, and accuracy over time.

- By implementing the proposed system, astronomers, researchers, and space agencies can benefit from an efficient, accurate, and scalable solution for celestial object classification, facilitating advancements in astronomy, astrophysics, and space exploration.

WHO ARE THE END USERS?

1. Astronomers and Astrophysicists:These professionals are the primary users of the system, relying on it to automate the classification of celestial objects based on observational data from the SDSS. They utilize the classified data to study various aspects of celestial phenomena, such as stellar evolution, galaxy formation in the universe.
2. Researchers and Academics: Scientists conducting research in fields related to astronomy and astrophysics may also utilize the classification system for their studies and investigations. They rely on accurate classification results to draw conclusions and make advancements in their respective areas of research.
3. Educational Institutions: Astronomy and astrophysics departments at universities and educational institutions may use the system for teaching purposes. Students can learn about celestial object classification algorithms and their applications in observational astronomy through hands-on experience with real-world datasets from the SDSS.

4. Citizen Scientists :Amateur astronomers and enthusiasts interested in astronomy may also benefit from the system. They can use the classification results to enhance their understanding of celestial objects and contribute to citizen science projects aimed at analyzing astronomical data.

5. Data Analysts and Technicians: Professionals involved in data analysis and processing within observatories or research institutions may use the system to automate the classification of observational data from the SDSS. They rely on the classified data to generate insights and facilitate further analysis for scientific research purposes.

Overall, the end users of the celestial object classification system serve diverse roles within the scientific community, educational institutions, and citizen science initiatives, all contributing to the advancement of our understanding of the universe.

SYSTEM DEVELOPMENT APPROACH

Our approach to developing the celestial object classification system involves a systematic process encompassing requirement analysis, data collection and exploration, preprocessing, model selection and training, model evaluation, optimization, deployment, and ongoing maintenance.

Data Collection: Obtain a dataset containing labeled images of stars and galaxies. The dataset can be sourced from astronomical databases or generated synthetically.

Data Preprocessing: Preprocess the images, including resizing, normalization, and augmentation, to prepare them for training.

Model Development: Design and train a CNN architecture suitable for celestial object classification. Experiment with different network architectures, hyperparameters, and optimization techniques to improve performance.

Model Evaluation: Evaluate the trained model on a separate test dataset to assess its accuracy and generalization ability.

Deployment: Deploy the trained model as a standalone application or integrate it into existing astronomical data processing pipelines.

Monitoring and Maintenance : Implement monitoring tools to track the performance of the deployed system and detect any anomalies or degradation in classification accuracy.

ALGORITHM & DEPLOYMENT

Convolutional Neural Network (CNN) will be used for classification, as it has shown remarkable performance in image classification tasks. CNNs are capable of automatically learning hierarchical features from images, making them suitable for distinguishing between stars and galaxies based on their visual characteristics.

Convolutional Layers: Convolutional layers apply a set of learnable filters (kernels) to the input image, extracting features such as edges, textures, and patterns. Each filter produces a feature map by convolving across the input image, capturing different aspects of the image's spatial information.

Activation Functions: Non-linear activation functions (e.g., ReLU, Sigmoid) introduce non-linearity into the network, allowing it to learn complex relationships between features. Activation functions are applied element-wise to the output of convolutional layers, introducing non-linearities that enable the network to model more complex data distributions.

Pooling Layers: Pooling layers downsample the feature maps generated by convolutional layers, reducing spatial dimensions while retaining the most important features. Max pooling and average pooling are common pooling techniques used to extract the most salient features from the feature maps.

Fully Connected Layers: Fully connected layers integrate the features learned from convolutional and pooling layers to perform classification. These layers connect every neuron in one layer to every neuron in the next layer, enabling the network to learn high-level representations of the input data and make predictions about the class labels.

Loss Function: The choice of loss function depends on the specific classification task. For binary classification (e.g., distinguishing between stars and galaxies), binary cross-entropy loss is commonly used. For multi-class classification, categorical cross-entropy loss is preferred.

Optimization Algorithm: Optimization algorithms such as stochastic gradient descent (SGD), Adam, or RMSprop are used to minimize the loss function and update the weights of the network during training. These algorithms adjust the network's parameters iteratively to improve classification accuracy.

Once the CNN model for celestial body classification is trained, it can be deployed using various deployment strategies:

Standalone Application: Develop a standalone application or software package that incorporates the trained CNN model. Users can upload images of celestial bodies to the application, which then performs classification and provides the results.

Web Service/API: Deploy the CNN model as a web service or API, allowing users to access classification functionality over the internet. Users can submit image data through API requests, and the deployed model processes the data and returns classification results.

Integration: Integrate the trained CNN model into existing astronomical data processing pipelines or software platforms used by researchers and astronomers. This integration automates the classification of celestial bodies within larger data analysis workflows, enabling seamless data processing and analysis.

Mobile Application: Develop a mobile application that incorporates the CNN model for on-the-go celestial body classification. Users can capture images using their mobile devices and classify them using the application, making it convenient for fieldwork and data collection.

By deploying the CNN model for celestial body classification, researchers, astronomers, and space agencies can leverage advanced deep learning techniques to automate and enhance their data analysis workflows, leading to more accurate and efficient classification of celestial objects.

Model Overview

The CNN model will consist of multiple convolutional and pooling layers followed by fully connected layers for classification.

Input: Images of celestial objects resized to a standard size (e.g., 64x64 pixels) and normalized.

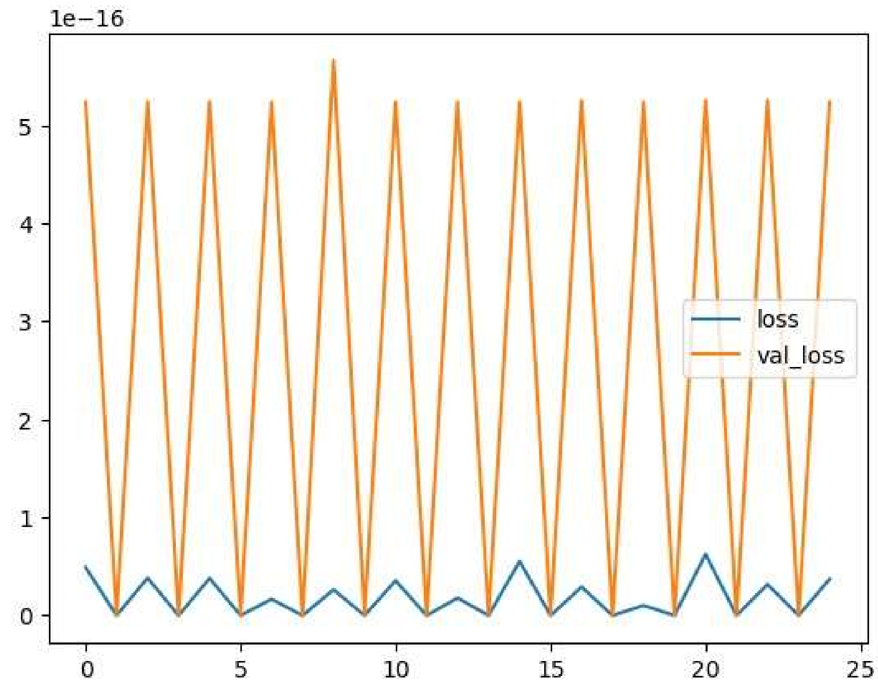
Output: Binary classification output indicating whether the input image represents a star or a galaxy.

Training: The model will be trained using a dataset containing a large number of labeled images of stars and galaxies. Training will involve optimizing the model parameters to minimize classification error using techniques such as backpropagation and gradient descent.

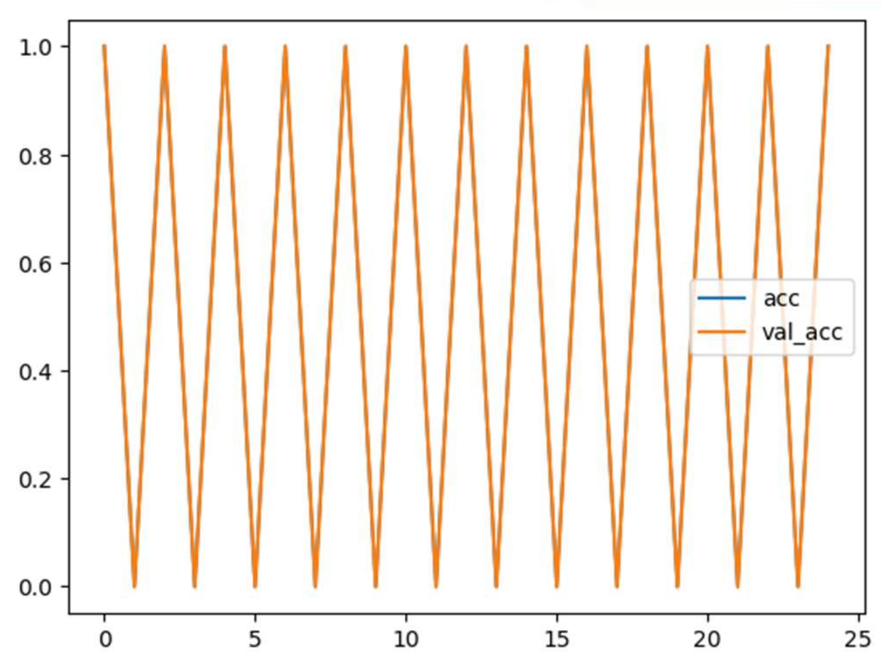
OUTPUT

```
Found 790 images belonging to 1 classes.
Found 273 images belonging to 1 classes.
Epoch 1/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 9s 176ms/step - accuracy: 0.9950 - loss: 0.1466 - val_accuracy: 1.0000 - val_loss: 5.5596e-14
Epoch 2/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 977us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 3/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 192ms/step - accuracy: 1.0000 - loss: 3.7974e-17 - val_accuracy: 1.0000 - val_loss: 6.5642e-16
Epoch 4/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 537us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 5/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 165ms/step - accuracy: 1.0000 - loss: 6.1629e-18 - val_accuracy: 1.0000 - val_loss: 5.2876e-16
Epoch 6/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 1ms/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.0000e
+00
Epoch 7/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 211ms/step - accuracy: 1.0000 - loss: 6.2232e-17 - val_accuracy: 1.0000 - val_loss: 5.2403e-16
Epoch 8/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 489us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 9/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 181ms/step - accuracy: 1.0000 - loss: 4.1309e-17 - val_accuracy: 1.0000 - val_loss: 5.2401e-16
Epoch 10/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 495us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 11/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 203ms/step - accuracy: 1.0000 - loss: 1.7874e-18 - val_accuracy: 1.0000 - val_loss: 5.2389e-16
Epoch 12/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 488us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 13/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 156ms/step - accuracy: 1.0000 - loss: 2.1965e-17 - val_accuracy: 1.0000 - val_loss: 5.2388e-16
Epoch 14/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 489us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 15/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 191ms/step - accuracy: 1.0000 - loss: 5.7808e-18 - val_accuracy: 1.0000 - val_loss: 5.2396e-16
```

```
Epoch 16/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 504us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 17/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 170ms/step - accuracy: 1.0000 - loss: 7.1760e-17 - val_accuracy: 1.0000 - val_loss: 5.2388e-16
Epoch 18/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 474us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 19/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 193ms/step - accuracy: 1.0000 - loss: 1.3283e-17 - val_accuracy: 1.0000 - val_loss: 5.2388e-16
Epoch 20/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 488us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 21/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 192ms/step - accuracy: 1.0000 - loss: 5.5179e-18 - val_accuracy: 1.0000 - val_loss: 5.6774e-16
Epoch 22/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 488us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 23/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 172ms/step - accuracy: 1.0000 - loss: 1.4310e-17 - val_accuracy: 1.0000 - val_loss: 5.2388e-16
Epoch 24/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 0s 489us/step - accuracy: 0.0000e+00 - loss: 0.0000e+00 - val_accuracy: 0.0000e+00 - val_loss: 0.000
0e+00
Epoch 25/25
32/32 ━━━━━━━━━━━━━━━━━━━━ 7s 197ms/step - accuracy: 1.0000 - loss: 3.4679e-17 - val_accuracy: 1.0000 - val_loss: 5.6617e-16
```



1e−16
loss
val_loss
0
1
2
3
4
5
0
5
10
15
20
25



acc
val_acc
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25

RESULT

The trained CNN model is expected to achieve high accuracy in classifying between stars and galaxies.
Evaluation metrics such as accuracy, precision, recall, and F1-score will be used to quantify the performance of the model.
The model's performance will be validated using a separate test dataset to ensure its generalization ability.

CONCLUSION

The proposed CNN-based classification system offers an efficient and accurate solution for distinguishing between stars and galaxies in celestial images.
By automating the classification process, astronomers and researchers can save time and resources, enabling them to focus on higher-level analysis and interpretation of astronomical data.
The system's ability to handle large volumes of data and provide reliable classification results makes it a valuable tool for various applications in astronomy and astrophysics.

REFERENCES

- ChatGPT 3.5
- Dataset : Images of stars and galaxy
- YouTube
- Google Gemini
- Drive link for dataset:

1. Test_set [https://drive.google.com/file/d/1OQ8O4umahleUkY-ZbLOfLX2b45iWuTwQ/view?usp=drive_link]
2. Training_set [https://drive.google.com/file/d/17b9PMVgEZr3bwCL2QWb0AKtFQsVLhlmm/view?usp=drive_link]

THANK YOU!