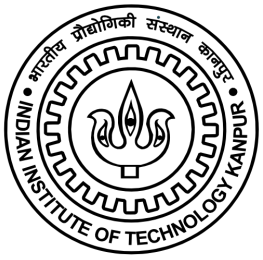
TAKNEEK 2024

HALL 2



KSHATRIYAS

QUEST 5

AstroExplorer

Overview

This project addresses the challenge of automating the classification and enhancement of astronomical objects using machine learning. Given the massive influx of data from modern telescopes and space missions, manual processing is no longer practical. The solution involves developing a tool that can classify various astronomical objects from FITS images and light curves. The tool not only identifies the object type but also applies specific image enhancement techniques tailored to each object.

The project comprises three main components:

1. **Data Collection and Dataset Generation**: Curating and processing datasets of astronomical objects, including both images and light curves, to be used for training and testing the model.
2. **Image and Light Curve Classification**: Building a machine learning model, specifically using CNN and DNN architectures, to accurately classify these objects. The model's performance is optimized through various techniques to achieve high accuracy.
3. **Image Enhancement**: Developing an algorithm that enhances the classified images by combining multiple spectral bands, improving visibility and signal-to-noise ratio based on the object type.

This automated approach significantly reduces the manual effort required in astronomical data analysis, making it a vital tool in modern astronomy.

1 Astronomical Object Details

1.1 Introduction

Astronomical objects, from stars and planets to galaxies and black holes, are the fundamental components of the universe. Each type of object has distinct characteristics—such as size, luminosity, composition, and motion—that allow scientists to classify and study them. Understanding these properties helps reveal the structure and evolution of the cosmos. This document explores the key features that differentiate various astronomical objects, providing insights into their unique roles in the universe.

1.2 Objects Being Classified

- **Galaxies**
- **Stars**
- **Quasars**

Galaxy

Information

- **Types of Galaxies**: Galaxies are categorised into four main types based on their shapes:
  - **Elliptical**: Round or elongated, with little gas and dust.
  - **Spiral**: Flattened disks with central bulges and spiral arms.
  - **Barred Spiral**: Similar to spiral galaxies, but with a central bar-shaped structure.
  - **Irregular**: No definite shape, often resulting from gravitational interactions or collisions.
- **Dark Matter**: A significant portion of a galaxy's mass is made up of dark matter, a mysterious substance that doesn't emit light but affects the galaxy's gravitational behaviour. Initially, without taking into consideration any existence of dark matter, the observed spiral arms of the galaxy could not be explained due to less total mass calculated, which was later resolved by taking into account the hypothetical "DARK MATTER."
- **Galaxy Clusters**: Galaxies are not isolated but exist in groups or clusters. These clusters can contain thousands of galaxies bound by gravity. A famous one is Laniakea Supercluster, also called the "Great Attractor," which attracts all the nearby smaller galaxy clusters.

- **Hubble's Law**: The universe is expanding, and galaxies are moving away from each other. The farther a galaxy is, the faster it's moving away, a relationship described by Hubble's Law.

Differentiating Factors

- **Gravitational lensing**: General Theory of Relativity shows that mass bends spacetime, which tends to bend light. The effect of this bending can be visible around massive objects like SMBHs also, at the centre of galaxies. This bending acts like a lens by bending light.
- **Relatively continuous spectrum**: Different celestial objects, like gas clouds, stars, quasars, etc. tend to have different discrete spectra, which when combined in a galaxy, tend to give a comparatively more continuous spectrum.
- **X-ray and radio emission**: The hot gas, supernova remnants, or black holes within the galaxy, having high energy emit X-rays. Radio emissions arise from active star formation regions, pulsars, or black holes.
- **Polarisation due to interstellar dust**: The interstellar dust present in the space between the galaxy and Earth can act as polarizers, polarising the light incoming from the galaxy.
- **Extreme points slight red-blue shift difference**: The part of the galaxy moving towards you will be relatively less red-shifted compared to the point moving away from us.
- **Central high luminosity peak due to active SMBH**: The galactic bulge at the centre of galaxies due to the SMBH (if present) has a much higher brightness than other parts of the galaxy due to the formation of an accretion disk by engulfing nearby stars and celestial objects.
- **Wider parallax**: For a given distance, compared to other celestial objects, galaxies have a wider parallax due to their huge size.
- **Linear motion along with internal rotational motion**: Along with the linear motion of galaxies moving away from us, internal rotational motion can also be seen in galaxies.
- **Can be elliptical**: Most of the galaxies being planar, can look elliptical when looked at an angle to the plane vector of the galaxy.

Stars

Information

Stars are massive, luminous spheres of plasma held together by gravity, sustained by nuclear reactions in their core. These celestial bodies are the building blocks of galaxies, influencing the universe's structure and evolution. The life cycle of a star begins with protostellar formation, followed by a main-sequence stage where hydrogen fusion occurs. As stars age, they undergo significant changes, such as expanding into red giants, fusing helium into heavier elements, and shedding their outer layer.

Stars are classified based on their spectral type (O, B, A, F, G, K, M), luminosity class (I-V), and metallicity. Their properties, including mass, radius, surface temperature, and composition, determine their evolutionary pathways. Advanced astrophysical processes, such as stellar rotation, magnetic fields, and asteroseismology, provide valuable insights into stellar interiors and atmospheres.

Throughout their life cycle, stars constantly struggle with the force of gravity. Gravity works to collapse the star, but the hot core creates pressure that counteracts this force, achieving what is known as hydrostatic equilibrium. A star remains stable as long as there is a balance between the inward pull of gravity and the outward push of pressure.

Differentiating Factors

- **Through Naked Eyes and Earth-Based Telescopes**: Stars generally appear as point-like and twinkling when viewed from Earth. They exhibit observable parallax, which can provide valuable information such as proximity from Earth and radius.

- **Light Curves**: Stars typically have a higher surface brightness (10-100 times brighter than their surroundings) due to their intense radiation.
- **Spectral Lines**: Stars have unique spectral lines that provide valuable data about their composition, age, and other properties. Narrow spectral lines with well-defined types (O, B, A, F, G, K, M) indicate surface temperature, atmospheric conditions, and variable phenomena like granulation.

Quasars

Information

A quasar is an extremely active and luminous type of active galactic nucleus (AGN). All quasars are AGNs, but not all AGNs are quasars. Quasars are the blazing centers of active galaxies and are powered by a supermassive black hole feeding on huge quantities of gas. The matter in the form of huge clouds falls into the disk, with the inner parts of the cloud closer to the black hole orbiting faster than the outer parts. This creates a shear force that twists the clouds, causing them to collide with their neighbors as they move around the black hole. This friction from fast-moving gas clouds generates heat, and the disk becomes so hot—millions of degrees—that it shines brightly.

Differentiating Factors

- **Through Naked Eye**: Quasars appear as point-like objects similar to stars but with different colors. They are variable sources of light, and their brightness changes over time, aiding in their identification.
- **X-ray and Radio Emission**: Quasars are strong emitters of radio waves, detectable in radio surveys. They are also strong X-ray emitters, and observatories like Chandra and XMM-Newton are used to identify them.
- **Colour Selection**: In optical bands, quasars appear blue due to excess ultraviolet radiation. In color-color diagrams, quasars occupy different regions compared to stars and galaxies.
- **Spectral Energy Distribution (SED)**: Quasars cover the full electromagnetic spectrum, whereas the SEDs of stars are confined to visible and infrared regions, and galaxies are less pronounced in X-rays and radio regions. Overall, quasars have distinct SEDs that differ from those of galaxies and stars.
- **Spectral Lines and Emission Spectra**: Quasars have strong and broad emission lines in their spectra, exhibiting a continuous spectrum with few or no absorption lines due to the high speed of gas particles.
- **High Luminosity**: Quasars are among the most luminous objects in the universe, with luminosities often 100 to 1000 times greater than a typical galaxy.

2 Image Classification ML Model

2.1 Introduction

We made a machine learning model whic is designed to classify astronomical objects such as stars, galaxies, and quasars, based on image data. The model utilizes a Convolutional Neural Network (CNN) architecture, which is particularly effective for processing and analyzing the complex, high-dimensional image data. The primary objective of this model is to accurately identify and classify various types of astronomical objects using their visual characteristics captured in images, which are cropped to 32x32 pixels with 5 channels.

The dataset employed in this model is derived from the Sloan Digital Sky Survey (SDSS) and includes a substantial collection of images, each associated with a label indicating the type of astronomical object it represents. To improve the model's robustness and generalization capabilities, advanced data augmentation techniques are applied during the training process.

This documentation provides a detailed overview of the model's architecture, data handling methodologies, training procedures, and the challenges faced during its development. Through meticulous design and optimization, this model aims to achieve high accuracy in classifying astronomical objects, thereby contributing valuable insights to the field of astronomy.

2.2 Libraries Used

The following libraries were utilized in the development and implementation of the machine learning models for classifying astronomical objects:

- **NumPy**
- **Pandas**
- **TensorFlow**
- **Keras (TensorFlow submodule)**
- **Scikit-learn**
- **Matplotlib**.

2.3 Dataset Description

The dataset used in this project consists of astronomical images and corresponding target labels:

- **Image Data (X):** The dataset `X` contains cropped images of astronomical objects, each with a resolution of 32x32 pixels and 5 channels. These 5 channels are (u, g, r, i, z) which are Ultraviolet, Green, Red, Infrared and All other remaining Spectrum. These images were obtained and processed to serve as input data for the machine learning models.
- **Target Labels (y):** The dataset `y` comprises categorical labels representing the classification of each astronomical object. These labels serve as the target output for the models during training.

2.3.1 Data Acquisition Details

The 5-band image data was obtained from the SDSS database. We downloaded the images from the website and were saved in a format of 32x32 pixels, and organized the data for further analysis.

2.4 Data Preparation

The data was prepared through the following steps:

1. **Initial Splitting**: The dataset was divided into training (50%) and test (50%) sets with reproducibility ensured by `random_state=42`.
2. **Further Splitting for Validation**: The training set was further split, reserving 15% for validation.

   This process ensures proper training, validation, and testing of the model.

2.5 Architecture of the Model

The architecture of the model is centered around a deep Convolutional Neural Network (CNN), specifically designed to classify astronomical objects from image data. The CNN model is constructed with multiple convolutional layers, activation functions, and pooling layers to extract high-level features from the input images. The model can be broken down into the following key components:

2.5.1 Input Layer

- The input layer accepts images of shape $32 \times 32 \times 5$, where each image is represented with five channels.

2.5.2 Convolutional Blocks

- The model begins with a series of convolutional blocks. Each block consists of convolutional layers with ReLU activation functions, designed to extract various features from the images.
- The initial convolutional layers use small 1×1 kernels, followed by layers with larger 3×3 and 5×5 kernels to capture both fine and coarse features.
- Average pooling layers are strategically placed within the network to reduce the spatial dimensions while preserving important features.

2.5.3 Inception-Like Module

- The architecture employs an inception-like module, where outputs from different convolutional filters are concatenated. This allows the model to capture multi-scale features and enhances its robustness.
- This module incorporates filters of various sizes and combines them to form a rich representation of the input data.

2.5.4 Fully Connected Layers

- After the convolutional layers, the model flattens the output and passes it through fully connected (dense) layers.
- Two dense layers with 1024 units each are included to perform high-level reasoning and feature abstraction.

2.5.5 Output Layer

- The final layer is a dense layer with a softmax activation function, producing a probability distribution over three classes. This layer is responsible for the final classification of the astronomical objects.

2.5.6 Compilation

- The model is compiled using the Adam optimizer, with categorical crossentropy as the loss function. The metric used for evaluating the model's performance is accuracy.

The overall architecture of the model is designed to leverage the strengths of CNNs in feature extraction while ensuring that the model can generalize well to different types of astronomical objects through robust training and regularization techniques.

2.6 Challenges Faced

The development and implementation of the Convolutional Neural Network (CNN) model for classifying astronomical objects presented several challenges. These challenges were addressed through careful planning, experimentation, and optimization. Below are some of the key challenges encountered:

2.6.1 Data Handling and Preprocessing

- **Data Format Compatibility:** The input data consisted of images with five channels, which required specialized handling during preprocessing. Ensuring that these images were correctly formatted for input into the CNN model posed a significant challenge.
- **Data Augmentation:** To enhance the generalization capability of the model, data augmentation techniques such as random cropping, rotation, and scaling were implemented. Balancing these transformations while maintaining the integrity of the astronomical features was a complex task.
- **Memory Management:** The dataset was large, with 240,000 samples, which strained the available computational resources. Efficient memory management strategies, such as batch processing and data generators, were crucial to prevent out-of-memory errors during training.

2.6.2 Model Design and Architecture

- **Architecture Complexity:** Designing an effective CNN architecture that could extract meaningful features from multi-channel images required extensive experimentation. The use of inception-like modules added complexity but was necessary to capture multi-scale features.
- **Overfitting and Underfitting:** Striking the right balance between model complexity and generalization was challenging. Techniques like dropout, regularization, and careful tuning of the network's depth and width were employed to mitigate overfitting, while ensuring that the model did not underfit the data.
- **Integration of Multiple Layers:** The architecture involved integrating layers with different kernel sizes and pooling strategies. Ensuring seamless integration of these layers without losing important spatial information was a non-trivial task.

2.6.3 Training and Optimization

- **Convergence Issues:** The training process encountered challenges related to slow convergence and local minima. Adjusting learning rates, implementing early stopping, and using techniques like learning rate schedules were necessary to ensure proper convergence.
- **Computational Resource Constraints:** Training the model on large datasets with a complex architecture demanded substantial computational resources. Leveraging GPUs and optimizing batch sizes were critical to manage training times effectively.
- **Hyperparameter Tuning:** Finding the optimal set of hyperparameters, such as learning rate, batch size, and the number of layers, required extensive experimentation. Grid search and random search methods were employed to systematically explore the hyperparameter space.

2.6.4 Model Evaluation and Validation

- **Model Evaluation Metrics:** Accurately evaluating the model's performance was challenging due to the imbalanced nature of the dataset. It was essential to go beyond accuracy and consider metrics like precision, recall, and F1-score to get a comprehensive understanding of the model's effectiveness.
- **Cross-Validation:** Implementing robust cross-validation techniques was necessary to ensure that the model's performance generalized well to unseen data. This involved splitting the data carefully to avoid any data leakage and ensure fair evaluation.

2.6.5 Deployment and Scalability

- **Model Integration:** Integrating the CNN model into a broader system posed challenges, particularly in ensuring that the model could be deployed efficiently and perform well in a real-time environment. This required optimization for latency and scalability.
- **Continuous Improvement:** Post-deployment, maintaining the model's performance through regular updates and retraining with new data was crucial. This required a well-structured approach to version control and model maintenance.

Overall, these challenges highlighted the complexity of developing a machine learning model for astronomical object classification. However, by systematically addressing each challenge, a robust and effective model was achieved.

3 Object Classification through Light Curves

3.1 Introduction

This documentation presents a machine learning model designed for classifying astronomical objects based on their light curves. The model utilizes Long Short-Term Memory (LSTM) networks, a type of recurrent neural network (RNN) well-suited for handling sequential data. Light curves, which represent the variation in brightness of an astronomical object over time, serve as the primary input for this classification task.

- **Classification Task:** The primary objective of this model is to accurately classify astronomical objects into predefined categories based on their light curves. Light curves are time-series data that reflect changes in the brightness of an object, which can provide valuable insights into the nature of the object. By analyzing these light curves, the model aims to differentiate between various types of celestial objects such as variable stars, eclipsing binaries, and other time-varying phenomena. The classification task involves training the model on labeled light curve data and enabling it to recognize patterns and anomalies that correspond to different object types. This task is crucial for enhancing our understanding of celestial objects and their behaviors.

3.2 Libraries Used

The following libraries were utilized in the development and implementation of the machine learning models for classifying astronomical objects:

- **NumPy**
- **Pandas**
- **TensorFlow**
- **Keras (TensorFlow submodule)**
- **Scikit-learn**
- **Matplotlib**.

3.3 Data Acquisition

The dataset used for classifying astronomical objects was downloaded from the Optical Gravitational Lensing Experiment (OGLE) website, available at `https://ogledb.astrouw.edu.pl/~ogle/OCVS/`. The data was then stored in a CSV file named `combined_light_curve_data.csv` which contains heading **label** for type of Star, headings **mjd** for time series data, heading **flux** for time-corresponding infra-red flux data and heading **flux_error** having error in the flux .

3.4 Data Description

The given csv file contains the data downloaded from the site and and then processed as needed. It contains five types of variable stars whic are **Sigma_SCT**, **Classical_Cepheids**, **Eclipsing**, **Miras** and **RR_Lyrae**. As the number of lightcurves were beyond the computational limit of our devices, we had to truncate the dataset to contain 7500 lightcurves for each category. These lightcurves contain varying amount of datapoints so we had to equalise them to a constant number which we took to be 350 points per light curve. We used padding and trucation to do so. As the csv we had created had stored the data points as a string array, we had to convert it into float and normalise the data.

We integer encoded the labels to represent points from 0 to 4. The label mapping is as follows:

- **Classical_Cepheids**: 0
- **Eclipsing**: 1
- **Miras**: 2
- **RR_Lyrae**: 3
- **Sigma_SCT**: 4

We then dropped the label and flux_error cloumn and replaced it with label_encoded column in the dataset csv file. We had to drop the flux_error column as it was not providing anything substantial to out LSTM model and also was challenging for our model as it was causing high computations.

3.5 Model Architecture

In this model, we implemented a Long Short-Term Memory (LSTM) network for the classification of variable star light curves. The model was constructed using the Sequential API from TensorFlow's Keras library, enabling a straightforward linear stacking of layers.

The architecture begins with an LSTM layer consisting of 128 units. This layer is designed to capture the temporal dependencies in the input sequences and is configured to return sequences, allowing the following layers to process the entire sequence of data. To mitigate the risk of overfitting, a Dropout layer with a rate of 0.2 is added after the first LSTM layer, which randomly sets 20% of the input units to zero during each update in the training phase.

A second LSTM layer, with 64 units, is then applied to further distill the sequence information. This layer does not return sequences, as it is the final recurrent layer in the network. Another Dropout layer, identical to the first, is included after the second LSTM layer to provide additional regularization.

The final output of the model is produced by a Dense layer, which uses a softmax activation function to classify the input sequences into one of the predefined classes. The number of units in this layer corresponds to the number of classes in the classification task.

3.6 Model Compilation

The model was compiled using the Adam optimizer, which is known for its efficiency and ability to adapt learning rates during training. Categorical crossentropy was employed as the loss function, suitable for multi-class classification problems. The model's performance was evaluated using accuracy as the primary metric.

3.7 Generation of Bi-Dimensional Histograms and Integration with LSTM Model

To facilitate the classification of variable star light curves, we first transformed the one-dimensional light curve data into bi-dimensional histograms, also known as dm-dt mappings. These histograms represent the distribution of magnitude differences (Δm) and time differences (Δt) between pairs of observations, converting the sequential temporal data into a format that can capture both temporal and magnitude variations.

Each light curve underwent a pre-processing step to remove any missing or invalid data points. Following this, the differences in magnitude and time were computed for consecutive observations within each light curve. These differences were used to construct a 2D histogram, where the x-axis represents the magnitude differences (Δm) and the y-axis represents the time differences (Δt).

Instead of applying these bi-dimensional histograms to a Convolutional Neural Network (CNN), we used them as a complementary feature set within a Long Short-Term Memory (LSTM) network framework. The LSTM model, known for its capability to learn and remember temporal patterns in sequential data, was applied directly to the original one-dimensional light curve sequences. The LSTM network was structured with two layers of LSTM units, followed by dropout layers for regularization, and a final dense layer with a softmax activation function for classification.

The integration of LSTM with bi-dimensional histograms provides a hybrid approach where the LSTM model processes the sequential nature of the original light curves, while the histograms supply additional spatial-temporal features that enhance the model's ability to differentiate between classes. By combining these methods, the model leverages the strengths of both temporal sequence learning and spatial feature extraction, leading to improved accuracy in classifying variable stars.

3.8 Challenges Faced in Model Development

The development of the variable star classification model involved several key challenges:

1. **Data Manipulation for CSV Creation:** Consolidating raw light curve data into a single CSV file required extensive preprocessing, including aligning timestamps and handling missing values to ensure data consistency.
2. **Bi-Dimensional Histogram Generation:** Transforming one-dimensional light curves into bi-dimensional histograms (dm-dt mappings) was complex, particularly in selecting appropriate bin sizes and removing invalid data points. Adjustments were necessary to accurately capture the patterns in different datasets.
3. **Flux Error Column Removal:** The flux error column in the raw data introduced noise, and its removal required careful reprocessing to maintain the integrity of the light curves, simplifying the input data for better model performance.
4. **Padding for LSTM Input:** Handling sequences of varying lengths involved padding shorter sequences, which had to be done carefully to avoid distorting the temporal patterns essential for LSTM learning.

3.9 Challenges Faced in Model Structuring and Optimization

During the structuring and optimization of the classification model for variable star light curves, several challenges were encountered:

1. **Hyperparameter Tuning:** Determining the optimal hyperparameters, such as the number of LSTM units and dropout rates, proved challenging. Extensive experimentation was required to find the right balance between model complexity and generalization to avoid overfitting while maintaining high performance.

2. **Overfitting:** The initial models showed signs of overfitting, where the model performed well on training data but poorly on validation data. This issue was addressed by incorporating dropout layers and regularization techniques to enhance generalization and improve the model's performance on unseen data.
3. **Computational Resources:** Training the LSTM network with large datasets and multiple layers was computationally intensive. Managing computational resources and training times required careful optimization and efficient use of hardware, such as GPUs, to handle the extensive computations involved.
4. **Data Variability:** The variability in light curve data from different sources introduced inconsistencies, which affected the model's performance. Preprocessing steps, such as normalization and handling missing values, were critical but challenging to implement effectively across diverse datasets.
5. **Sequence Padding:** Handling sequences of varying lengths required padding shorter sequences, which had to be done carefully to prevent distortion of temporal patterns. Proper padding strategies were crucial to maintain the integrity of the data and ensure effective learning by the LSTM network.

References

- Saksham Bassi, Kaushal Sharma and Atharva Gomekar (Classification of various Star Light Curves Using Long Short Term Memory Network)
- Shuying Liu, Weihong Deng Beijing University of Posts and Telecommunications, Beijing, China(Very Deep Convolutional Neural Network Based Image Classification Using Small Training Sample Size)
- Dan C. Cires¸an, Ueli Meier, Jonathan Masci, Luca M. Gambardella, Jurgen Schmidhuber (Flexible, High Performance Convolutional Neural Networks for Image Classification)
- Daniele L.R. Marini, Cristian Bonanomi, Alessandro Rizzi; Università degli Studi di Milano, Dipartimento di Informatica; Milano, Italy(Processing astro-photographs using Retinex based methods)
- ChatGPT