The user wants me to translate a Chinese academic paper title and subtitle. The title is "Ensemble Learning for Stellar Classification and Radius Estimation from Multimodal Data" and the subtitle is "后印本". Let me break this down: - "Ensemble Learning" is already in English, so I should keep it as···

Authors: The user has provided a list of names: "Zhi-Jie Deng, Sheng-Yuan Yu, A-Li Luo, Xiao Kong and Xiang-Ru Li"

This appears to be author names for a scientific paper. The task is to translate from Simplified Chinese to English with the highest accuracy and academic tone.

Let me analyze the requirements: 1. Preserve ALL LaTeX commands and placeholders - there are none here 2. Preserve ALL citation commands - there are none here 3. **PRESERVE ALL ···paragraph wrapper tags EXACTLY** - there are none here 4. Maintain academic tone and formal scientific writing style 5. Use precise technical terminology - obey the glossary strictly 6. Preserve section structure and paragraph organization 7. Translate all content completely - do not omit any information

The names are already in a format that is standard for English academic papers (Pinyin transliteration). The only thing I might consider is whether "A-Li Luo" should be "A-Li Luo" or "Aliluo" or something else, but "A-Li Luo" is the correct Pinyin representation. Similarly, all other names are already in proper Pinyin format.

The glossary provided is: - 机器学习 => machine learning - 深度学习 => deep learning

But these terms don't appear in the input.

The input is just a list of names separated by commas and "and". This is typical for author lists in academic papers. In English, this format is already correct. The names are: - Zhi-Jie Deng (邓志杰) - Sheng-Yuan Yu (余盛源) - A-Li Luo (罗阿丽) - Xiao Kong (孔晓) - Xiang-Ru Li (李祥儒)

Since the names are already in Pinyin, which is the standard way to represent Chinese names in English academic publications, no translation is needed. The format is also correct for an author list.

However, I need to check if there are any tags. The user instruction says "PRESERVE ALL ···paragraph wrapper tags EXACTLY". But the input doesn't contain any such tags. This is just a simple line of text.

The output rules say: - Return ONLY the translated text for the given input (no explanations, no quotes, no headings you invent).

Date: 2024-12-06T00:00:00+00:00

Abstract

The user wants me to translate a Chinese academic text to English. Let me break down the requirements:

1. **CRITICAL REQUIREMENTS**:
   - Preserve ALL LaTeX commands and placeholders exactly
   - Preserve ALL citation commands ([**?**], **??**, (**??**), etc.) exactly
   - **PRESERVE ALL ···paragraph wrapper tags EXACTLY**
   - Maintain academic tone and formal scientific writing style
   - Use precise technical terminology (glossary provided)
   - Preserve section structure and paragraph organization
   - Translate all content completely - no omissions
2. **OUTPUT RULES**:
   - Return ONLY the translated text (no explanations, quotes, or invented headings)
   - Keep one output paragraph per input paragraph
   - **Maintain exact count and IDs of ···tags**
   - Do NOT add Markdown formatting unless present in source
   - Preserve original line breaks when meaningful
3. **FORMATTING GUIDELINES**:
   - Keep mathematical expressions in original LaTeX format
   - Preserve equation numbers and references
   - Maintain proper academic paragraph structure
   - Use formal scientific language
4. **Glossary**:
   - 机器学习 => machine learning
   - 深度学习 => deep learning

Now, let me analyze the input text. The user provided a single paragraph of Chinese text without any tags, LaTeX commands, or mathematical formulas. This means I need to translate it as a single paragraph and wrap it in a ···tag since the instructions say "if input has N paragraph tags, output must have exactly N paragraph tags with matching IDs". Since there are 0 tags in the input, but the example shows the system expects tags, I should probably add

one tag.

Wait, let me re-read the instructions more carefully. The user says "Translate from Simplified Chinese to English" and provides text. The text doesn' t contain any tags. But the instructions say "PRESERVE ALL ···paragraph wrapper tags EXACTLY" . This is a bit ambiguous.

Looking at the example format in the instructions, it seems like the system expects the input to be wrapped in tags. Since the user' s input is not wrapped, I have two options: 1. Just translate the text as is without any tags 2. Wrap the translated text in ···

Given that the instructions emphasize preserving these tags and maintaining exact counts, and since the input has 0 tags, the most logical interpretation is that I should output 0 tags as well - just the translated text. However, the system seems designed around these tags being present. Let me check the instructions again.

The instructions say: "**PRESERVE ALL ···paragraph wrapper tags EXACTLY** - these are structural markers that identify paragraphs and MUST remain untouched with their IDs" . This suggests that if they exist, I must preserve them. If they don' t exist, I don' t need to add them.

But then it says: "**Maintain exact count and IDs of ···tags** - if input has N paragraph tags, output must have exactly N paragraph tags with matching IDs" . This

Full Text

Preamble

Research in Astronomy and Astrophysics, 24:115019 (14pp), 2024 November

https://doi.org/10.1088/1674-4527/ad86a6

Ensemble Learning for Stellar Classification and Radius Estimation from Multimodal Data

Zhi-Jie Deng[1], Sheng-Yuan Yu[2], A-Li Luo[3],[4], Xiao Kong[3], and Xiang-Ru Li[1]
[1] School of Computer Science, South China Normal University, Guangzhou 510631, China; lixiangru@scnu.edu.cn
[2] School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518055, China
[3] CAS Key Laboratory of Optical Astronomy, National Astronomical Observatories, Chinese Academy of Sciences, Beijing 100101, China; lal@nao.cas.cn
[4] School of Astronomy and Space Science, University of Chinese Academy of Sciences, Beijing 101408, China

Received 2024 April 30; revised 2024 August 14; accepted 2024 August 15; published 2024 November 14

Abstract

Stellar classification and radius estimation are crucial for understanding the structure of the Universe and stellar evolution. With the advent of the era of astronomical big data, multimodal data are available and theoretically effective for stellar classification and radius estimation. A key challenge is how to improve the performance of these tasks by jointly using multimodal data. However, existing research primarily focuses on using single-modal data. To address this limitation, this paper proposes a model called Multi-Modal SCNet (MM-SCNet) and its ensemble model Multimodal Ensemble for Stellar Classification and Regression (MESCR) to improve stellar classification and radius estimation performance by fusing data from two modalities. A typical phenomenon in this problem is that the sample numbers of some stellar types are evidently larger than others. This imbalance has negative effects on model performance.

Therefore, this work utilizes a weighted sampling strategy to address the imbalance issues in MESCR. Evaluation experiments conducted on a test set show that MESCR achieves a classification accuracy of 96.1%, with radius estimation performance yielding a Mean Absolute Error (MAE) of 0.084 dex and a standard deviation (σ) of 0.149 R , respectively. Moreover, we assessed the uncertainty of model predictions, confirming good consistency within a reasonable deviation range. Finally, we applied our model to 50,871,534 SDSS stars without spectra and published a new catalog.

Key words: methods: data analysis –techniques: image processing –methods: statistical

1. Introduction

In recent years, various large-scale sky survey projects have been continuously carried out, such as the Sloan Digital Sky Survey (SDSS; [**?**, **?**]), Gaia ([**?**]), LAMOST ([**?**]), and the Large Synoptic Survey Telescope (LSST; [**?**]). These projects have provided us with an unprecedented amount of data, enriching our understanding of the distribution of stars in the Universe, their physical properties, and evolutionary processes. To conduct astronomy research based on these big data, two essential tasks are to classify observed stellar data and estimate their parameters.

Based on the Morgan-Keenan classification method ([**?**]) and surface temperature, stars are classified into O, B, A, F, G, K, and M types from hot to cold. Each category is further divided into ten subclasses (0-9) based on decreasing surface temperature. Historically, this classification task was conducted by astronomers through visual inspection. However, this approach is infeasible in the era of big data, necessitating the development of automated classification

methods such as template matching (e.g., [**?**]) and machine learning methods (e.g., [**?**]).

The stellar radius is an important parameter for understanding stellar morphology. Methods for measuring stellar radius can be divided into direct and indirect types. Direct methods include interferometry and eclipsing binary fitting. Interferometry allows astronomers to directly measure the angular diameter of stars and calculate their actual radius by combining known distances ([**?**, **?**]). The eclipsing binary fitting method obtains physical information about stars by analyzing the light curve of eclipsing binaries ([**?**, **?**]). Indirect methods include deriving radius by analyzing stellar brightness, temperature, and distance, or using empirical relations based on observational data for fitting (e.g., [**?**]). However, the accuracy of these methods is limited by the models and assumptions they rely on, and error accumulation may increase result uncertainty. In recent years, machine learning methods have become increasingly popular for automatically estimating stellar parameters (e.g., [**?**]).

The machine learning methods mentioned above typically use data from a single modality for classification or parameter estimation. With large-scale sky survey projects continuing to develop, we can now obtain data from different bands and instrument types. This diversity provides more comprehensive and richer information. However, integrating data from multiple modalities to make information from different data types complementary and achieve higher accuracy has become a key challenge. Fortunately, some research has focused on this area (e.g., [**?**, **?**, **?**]), though no efforts have yet been made specifically for stellar classification and radius estimation.

Therefore, this work focuses on establishing a machine learning model to improve stellar classification and radius estimation performance through multimodal fusion.

The structure of this paper is as follows: Section 2 discusses the data acquisition process, Section 3 presents the principles of the proposed MESCR solution, Section 4 evaluates model performance through related experiments, Section 5 applies MESCR to SDSS photometric images to generate a new catalog, and Section 6 concludes the paper.

2. Data

In this study, the neural network accepts parallax and proper motion along with photometric images as inputs, simultaneously performing stellar classification and radius estimation. Proper motion, parallax, and stellar radii can be obtained from Gaia, while photometric images and spectral types can be obtained from SDSS. Therefore, the reference set was obtained by cross-matching the SDSS Data Release (DR) 17 catalog and the Gaia DR3 catalog. Subsequent sections detail the process of establishing the reference dataset and relevant data preprocessing steps.

2.1. Reference Dataset Based on Common Observations from SDSS DR17 and Gaia DR3

The Sloan Digital Sky Survey (SDSS) is one of the largest optical sky survey projects to date, dedicated to creating a detailed 3D map covering about one-third of the sky. SDSS uses CCD filters to collect photometric images in five optical bands: u, g, r, i, and z. The photometric images and spectra from SDSS can be accessed online via the Casjobs server.[1] [**?**] selected a subset of objects from SDSS DR17 by making the sample numbers of various stellar types as equal as possible, establishing a novel catalog (the Shi catalog). The type balance in the Shi catalog is helpful for training machine learning models.

[**?**] established their catalog by cross-matching the SDSS DR17 SpecPhoto and SpecObj catalogs. To ensure data quality, they further filtered out targets fainter than the limiting magnitudes, compiling a dataset of 46,245 stars. They used this catalog for stellar classification and achieved an accuracy of 86.1%. However, the Shi catalog does not include parallax, proper motion, or radius —all of which are necessary for this study. Therefore, this work established a reference set of common observations from SDSS DR17 and Gaia DR3 by cross-matching the Shi catalog with Gaia DR3.

The Gaia sky survey provides astrometric, photometric, and spectroscopic samples of nearly 2 billion stars in the Milky Way, as well as important samples of extragalactic and solar system objects. The Gaia Archive offers a rich dataset that includes calculated positions, parallaxes, proper motions, radial velocities, and brightness measurements of stars and other celestial bodies, accessible online via the Gaia Archive server.[2] Our dataset was constructed from Gaia DR3 ([**?**]). The parallax and proper motion used in this paper are from the Gaia source catalog, whereas the stellar radius (radius_{flame}) is from the astrophysical parameters catalog. For stellar radius, we choose to use the value calculated from the Stefan–Boltzmann law, derived from the star' s temperature and luminosity. The luminosity is obtained from the Final Luminosity Age Mass (FLAME) estimator using G-band magnitude, extinction, distance, and a bolometric correction ([**?**]).

The extinction calculation is performed by the GSP-Phot module, which analyzes BP and RP spectral data in conjunction with the star' s apparent G magnitude and parallax. This analysis is based on model fitting from the Aeneas best library that achieves the highest goodness-of-fit value. The GSP-Phot module uses the Markov Chain Monte Carlo (MCMC) method to obtain the median value of multiple samples to calculate the extinction value in the G band ([**?**]).

Due to some bright sources from Gaia that may saturate SDSS, we used the flags from SDSS DR17 to filter out saturated sources in the Shi catalog before performing the cross-match. Then, we cross-matched the Shi catalog with

[1] http://casjobs.sdss.org/casjobs/
[2] https://gea.esac.esa.int/archive/

the Gaia source and astrophysical parameters catalogs based on R.A. and decl. through the Gaia Archive server, with a cross-match radius of 3″. However, upon examining the radius distribution after cross-matching, we found that the radii of O and B-type stars did not match the sample. Despite further cross-matching with the SIMBAD astronomical database ([**?**]), the errors remained significant. To reduce data errors, we decided to exclude O and B-type stars from the dataset and limit our research to A, F, G, K, and M-type stars.

To further refine the dataset and obtain a more reliable catalog, we applied the following criteria to the cross-matched catalog to balance data quality and quantity: (1) parallax_{over}_{error} > 2 to select more accurate parallax data; (2) pmra_{error} < 0.2 and pmdec_{error} < 0.2 to exclude proper motion data with large errors; and (3) ruwe < 1.4 to exclude potentially problematic observational data, such as binaries or other complex background interferences.

[Figure 1: see original paper] shows the distributions of the used samples in (a) the parallax space, (b) the proper motion space, (c) the radius_{flame} space, and (d) the color–magnitude diagram (CMD). The CMD indicates that the data primarily consists of main sequence stars. In Figure 1(d), Gaia stars are used as a background, with the size of the points representing the signal-to-noise ratio (larger points indicate higher signal-to-noise ratio).

Our final dataset consists of 9,180 stars: 3,368 K-type stars, 2,918 F-type stars, 1,350 G-type stars, 918 A-type stars, and 626 M-type stars. Their distributions are presented in Figure 1. It is worth noting that there are relatively more K and F-type stars than G, A, and M-type stars, resulting in an imbalanced dataset. This poses challenges for model learning, with detailed solutions provided in Section 3.2.2. Moreover, we further processed the photometric images as described below.

2.2. Photometric Image Preprocessing

The photometric images obtained from SDSS DR16 are typically uncropped, high-resolution images that contain substantial irrelevant information, such as background noise, other celestial bodies, or blank areas. Therefore, we first use the World Coordinate System (WCS) to locate the target star and preliminarily crop the image around the star center to a size of 64×64 pixels. This greatly reduces irrelevant information while ensuring that the target star portion of the image is not cropped, preventing information loss. The photometric images include five bands: g, r, i, u, and z. To maximize the utilization of each band' s imagery, we repetitively use the r-band images to divide the five bands into two groups, gri and urz ([**?**]). Following the method described by [**?**], the cropped images from the gri and urz bands are converted into RGB images, as shown in Figure A1. They are subsequently referred to as gri and urz images, respectively.

To reduce irrelevant information in the images, we perform more refined cropping on the gri and urz images. The algorithm starts expanding the Region of Interest (ROI) from the image center to check if it contains all information of

the star. It calculates the position and size of each star in the r-band photometric image and then crops the gri and urz images accordingly. To implement this algorithm, we first define the detection area side length l, pixel intensity threshold α, and invalid pixel ratio threshold β. When the pixel intensity of a point in the image is less than or equal to α, we call this pixel "invalid" ; otherwise, it is "valid." For each 64×64 r-band image, we initially set a detection ROI of size $l \times l$ at the center of the image, as shown on the left of Figure A2, and calculate the ratio of invalid pixels to total pixels within this area. If this ratio exceeds β, we expand the side length and continue detection. Otherwise, the current ROI area is the target area, as shown on the right of Figure A2. Based on multiple experimental results, we set the initial ROI side length $l = 6$, intensity threshold $\alpha = 0.05$, and invalid pixel ratio threshold $\beta = 0.2$. To improve preprocessing efficiency, the algorithm only needs to calculate the newly added annular detection area in each iteration. After refined cropping, most image pixels fall within the 10–20 pixel range.

To standardize the image size for neural network learning, we resize the cropped images to 64×64 pixels using bilinear interpolation. We recognize that scaling from such a small size to 64×64 may introduce some degree of detail loss and artifacts. This issue will be further discussed in Section 4.3.2.

3. Method

A neural network is a hierarchical computational model consisting of a series of computing units called neural cells. The characteristic of a neural network is its ability for automatic feature learning ([**?**, **?**]). This work developed a model called Multi-Modal SCNet (MMSCNet) and its ensemble model Multimodal Ensemble for Stellar Classification and Regression (MESCR) to improve the accuracy of stellar classification and radius prediction by fusing multimodal data.

The core of this scheme is to improve overall performance by integrating the outputs of multiple sub-models as the final result through weighted sampling and ensemble learning. The following sections detail the network architecture of MMSCNet, MESCR, and how to use MESCR to achieve the final prediction objectives.

3.1. MMSCNet

The structure of MMSCNet is shown in Figure 2 [Figure 2: see original paper]. MMSCNet is composed of three main parts: an image feature extraction module, a numerical feature extraction module, and a multimodal feature fusion module. Each module is introduced in turn below.

In the image feature extraction module, we selected SCNet ([**?**]) as the backbone network due to its superior performance in stellar classification. SCNet integrates the CBAM attention mechanism ([**?**]) and is divided into two feature extraction stages. The first stage simultaneously accepts images from the gri

and urz bands as inputs, while the second stage stacks the feature maps from both branches for further feature extraction. The numerical data inputs consist only of parallax and proper motion. To avoid overfitting while extracting features from this data, we used two dense layers for the numerical data feature extraction module.

In the multimodal feature fusion module, the model concatenates the processed image feature vectors with the numerical feature vectors, forming a cross-modal feature vector. After concatenation, dense layers with softplus and linear outputs provide the scores for each class, stellar radius, and the corresponding uncertainty.

3.2. Multimodal Ensemble for Stellar Classification and Regression

As previously mentioned, this study considers a comprehensive task involving both classification and regression. In regression tasks, to derive the uncertainty of predictions and accurately estimate the probability density functions of physical parameters, deep ensemble methods are often required ([**?**, **?**]). In the classification tasks of this study, there exists an imbalance in the dataset, with the number of K and F-type stars significantly exceeding that of G, A, and M-type stars. This presents a challenging problem in machine learning ([**?**]). Numerous studies have proposed solutions to address such imbalanced classification tasks, including resampling techniques (e.g., [**?**]), cost-sensitive methods, adjusting algorithm mechanisms ([**?**]), and ensemble methods. Resampling techniques are widely used due to their universality and effectiveness. Ensemble methods not only have significant effects in solving data imbalance problems (e.g., [**?**, **?**]) but also match the requirements of the regression tasks in this study. Therefore, we use methods based on resampling techniques and ensemble methods as the basis for MESCR.

Specifically, we first perform repeatable weighted sampling on the imbalanced dataset to construct multiple sub-datasets, then adopt an ensemble learning method similar to bagging ([**?**]). The following two subsections introduce the process of constructing the dataset and the overall framework of MESCR, respectively.

3.2.1. Weighted Sampling and Data Augmentation Weighted sampling is a probability sampling technique in which each sample is assigned a weight that reflects its probability of being selected. Typically, the inverse of the sample frequency is used as the weight, meaning that minority class samples receive more attention. We first replicate the dataset six times, corresponding to datasets for five different stellar classes ($D_A, D_F, ..., D_M$) and one dataset that does not favor any class (D_{base}). The initial weight of a sample is set as the inverse of its class frequency. The weights in the D_{base} dataset remain unchanged, while for dataset D_i targeting a specific class, the weight of samples within that class is squared to further emphasize them: $w_i = w_i^2$. Subsequently, the weights of all samples are normalized to form the final sampling probability

distribution: $P(x_i) = w_i / \sum w_j$, where w_i is the adjusted weight of the sample and the denominator is the sum of all adjusted sample weights.

This method gives additional attention to rare classes in an imbalanced dataset, increasing the probability of these class samples being selected. During sampling, the weight of minority classes is further increased due to the squaring effect, while the weight of majority classes, which are already small, does not lead to more severe data imbalance after being squared. Ultimately, we obtain five datasets biased toward their respective classes ($D_A, D_F, ..., D_M$) and one relatively balanced dataset D_{base}. However, this repeatable sampling operation produces duplicate samples in our dataset, potentially leading to overfitting. Therefore, we perform five types of data augmentation on randomly selected data, including rotation, flipping, adding Gaussian noise, cutout, and erase, with actual effects shown in Figure A3. During training, data that has been sampled repeatedly will undergo data augmentation at least once, while other data will be randomly augmented with a certain probability. This method increases the diversity of the data.

To validate the model' s performance under real-world imbalanced data distribution, we performed repeatable weighted sampling only on the training and validation sets, without any processing on the test set. We first divided the original dataset into a training set and a test set in an 8:2 ratio, resulting in a training set with 7,344 samples and a test set with 1,836 samples. Subsequently, we replicated the training set six times. For each copy, we applied the aforementioned weighted sampling and at least one data augmentation method for each image, increasing the number of samples in each training set to 9,000. Finally, we divided this enhanced training set again in an 8:2 ratio to obtain a training set with 7,200 samples and a validation set with 1,800 samples for subsequent model training. The distribution of stellar spectral types in the resulting training, validation, and test sets is shown in Figure A4, while the distribution of proper motion, parallax, and radius_{flame} is shown in Figure A5.

3.2.2. MESCR Overall Framework and Training The proposed MESCR scheme is an ensemble learning strategy akin to bagging, consisting of six identical sub-models that use different weights in the sampling methods to construct the datasets. During training, we apply the five data augmentation methods mentioned in Section 3.2.1 to the training samples. The final output is divided into two parts: classification results and regression results. The classification results are obtained by averaging the outputs of the six models, with weights determined by the models' performance on the validation set. Through experimentation, we set the weights as follows: {Model Base: 0.1, Model A: 0.2, Model F: 0.1, Model G: 0.2, Model K: 0.15, Model M: 0.25}.

The ensemble stellar radius $\hat{x}_a$ is determined by the average prediction of the six models, and the ensemble uncertainty $\hat{x}_s^2$ is calculated using the formula: $\hat{x}_s^2 = \frac{1}{6}\sum_{i=1}^{6}(x_{a,i} - \hat{x}_m)^2 + \frac{1}{6}\sum_{i=1}^{6} x_{s,i}^2$, where $x_{a,i}$ and $x_{s,i}^2$ represent the predicted stellar radius and its corresponding uncertainty from the ith MMSCNet,

respectively, and $\hat{x}_m$ is the mean of the stellar radii obtained by the six sub-models.

For training each sub-model, we adopted the same strategy. We initialized each model with random weights to increase diversity among the models. We use two different loss functions for stellar classification and radius estimation, respectively. For stellar classification, we use the cross-entropy loss function. For stellar radius estimation, to enable the model to predict the probability density distribution, we use the negative log-likelihood of the normal distribution ([**?**]) as the model' s loss function: $-\log p(y|x) = \frac{1}{2}\log(2\pi\sigma_\theta^2(x)) + \frac{(y-\mu_\theta(x))^2}{2\sigma_\theta^2(x)}$, where x and y represent the input data and corresponding reference labels, respectively, $\mu_\theta(x)$ and $\sigma_\theta^2(x)$ represent the mean and variance of the Gaussian distribution predicted by the model, and θ are the parameters of the MMSCNet model to be optimized.

We adopted the following hyperparameters: batch size set to 256; total training epochs set to 200; baseline learning rate set at 0.001, with learning rate updated using the cosine annealing algorithm, performing a warm-up for the first three epochs, with the final learning rate set to 10^{-5}; using the SGD optimizer to optimize model parameters. The deep learning framework used was PyTorch, with an RTX 4060ti GPU. Training each sub-model took approximately 1 hour, while training the entire MESCR took about 6 hours in total.

Figure A6 shows the changes in loss values for each model of MESCR during the training process. As can be seen from the figure, the loss values for each model on both the training and validation sets gradually decrease with increasing iterations, indicating that the models' fitting and generalization capabilities improve progressively as training progresses.

4. Results and Discussions

This section evaluates the performance of the proposed model (Section 4.1) and investigates stellar radius estimation uncertainty (Section 4.2). Furthermore, we studied the impact of image band combinations and resolutions (Section 4.3), and assessed the improvements from incorporating parallax and proper motion (Section 4.4).

4.1. Model Evaluation

After selecting the optimal hyperparameters and completing training, this section uses the test set to comprehensively evaluate the predictive performance of MESCR. To explore the model' s classification and regression performance separately, we use different evaluation metrics.

4.1.1. Classification Result Evaluation We evaluate the overall performance using accuracy and measure the model' s ability to predict different categories using recall, precision, and F1 scores. Detailed definitions can be

found in [**?**]. MESCR achieved an overall accuracy of 96.1% on the test set. Table 1 (left) shows the F1 scores, recall, precision, and corresponding counts for MESCR' s predictions on the test set.

Figure 3(a) [Figure 3: see original paper] shows the confusion matrix of MESCR' s predictions on the test set. The numbers on the main diagonal represent correct predictions, while other non-zero positions indicate incorrect predictions. From Figure 3(a), most predictions lie on the main diagonal, indicating good classification ability. As for misclassified samples, they are mainly distributed in adjacent classes. For example, in samples where F and G-type stars were misclassified, all were classified into adjacent categories. We performed a homologous match between the SDSS samples described in Section 2.1 and LAMOST DR9, obtaining a test set containing 768 stars. We found that this phenomenon also occurs between SDSS and LAMOST, where an A-type star in SDSS may be classified into an adjacent category (B or F) in LAMOST. We compared MESCR prediction results with LAMOST spectral classification results, achieving an accuracy of 89.7%. The corresponding confusion matrix is shown in Figure 3(b), and the F1-score, recall, and precision are presented in Table 1 (right). Although our training samples are from SDSS, the accuracy of each metric remains within an acceptable range when using LAMOST spectral classification results as the reference. Additionally, as shown in Figure 3(b), most misclassified results are categorized into adjacent classes, indicating that classification into adjacent classes is normal.

4.1.2. Regression Result Evaluation We use the standard deviation (σ) between MESCR predictions and reference values and the Mean Absolute Error (MAE) as the main evaluation metrics. Detailed definitions can be found in [**?**]. The resulting MAE is 0.084 dex, σ is 0.149 R , and the mean predicted 1σ uncertainty is 0.016 R .

The lower subplot in Figure 4(a) [Figure 4: see original paper] shows a 1:1 scatter plot of predicted versus reference values, while the upper subplot shows a histogram of the distribution of differences between predicted and reference values. From the upper subplot, most differences are distributed in the range of $[-0.5, 0.5]$, indicating that the deviations between the model' s predictions and actual values are small. In the scatter plot below, the red dashed line represents ideal prediction where predicted values exactly equal true values. Most data points are concentrated in the lower left corner and close to the main diagonal, indicating that the model performs well in most cases. Overall, the model' s predictions closely match the reference values, and the high accuracy in reconstructing star properties demonstrates strong performance.

4.2. Uncertainty Analysis of Stellar Radius

This study employs a deep ensemble approach for predicting stellar radii and can provide corresponding predictive uncertainties σpred. Moreover, MESCR consists of six sub-models, each trained with a different training set and random

initial weights, ultimately yielding six distinct prediction results {y1, y2, ···, y6}. We calculate the standard deviation among these six prediction results as the model' s uncertainty, σnet.

Figure 4(b) shows the variation in uncertainty within different r-band SNR intervals. The dots represent the average uncertainty provided by MESCR for each SNR interval, serving as the predictive uncertainty σpred. The line segments centered on the circles denote the average standard deviation among the six sub-model results within each SNR interval, serving as the model' s uncertainty σnet. Overall, MESCR' s uncertainty is generally at a low level. As the S/N increases, both σnet and σpred decrease. When S/N $>$ 30, σpred stabilizes around 0.015 R . This indicates that the model performs better with higher S/N data, and its predictive ability is more stable for medium to high S/N data.

4.3. The Impact of Image Bands and Resolution on Model Performance

This section tested the model' s performance using different image resolutions and various band combinations separately. In these experiments, we tested each band combination and resolution using a single MMSCNet, retraining with the Dbase dataset from Section 3.2.1.

4.3.1. The Influence of Image Bands on Model Performance This work employs the r-band image reuse strategy, inputting gri and urz bands into the model. However, we aim to explore if there are more optimal combination schemes to assess the performance of various band combinations for specific tasks. Therefore, we selected the following band combinations for detailed analysis: [gri + ruz, gri + iuz, gri + guz, gru + giz, gru + riz, gru + iuz, giu + grz, giu + riz, giu + ruz, giz + riu, giz + ruz, guz + riu, guz + riz, iuz + grz, riu + grz]. These combinations all involve reusing one band to form a new combination. In previous work, [**?**] created three-channel images by averaging two channels (for example, the blue and red channels correspond to g and r-bands, respectively, while the green channel is the average of the two). Hence, we also adopted a similar approach to test the following band combinations: [gri + uz, gru + iz, grz + iu, giu + rz, giz + ru, guz + ri, riu + gz, riz + gu, ruz + gi, iuz + gr].

The experimental results in Table A1 clearly show the significant impact of different band combinations on model performance. In terms of stellar classification, accuracy fluctuated from a low of 92.7% (with the giz + ru combination) to a high of 95.7% (with the gru + iuz combination), indicating a potential performance improvement of up to 2.4%. Similarly, the MAE in regression tasks showed significant variability, from the best 0.077 (with the guz + ri combination) to the worst 0.087 (with the guz + riu combination), a difference of nearly 11.5%. These results confirm the decisive role of band selection for model efficacy.

4.3.2. The Influence of Image Resolution on Model Performance

Next, we fixed the band combination to gri + urz to explore the impact of image resolution on model performance. The experiment investigated various resolution settings ranging from 8 × 8 pixels to 224 × 224 pixels. We used bilinear interpolation to enlarge images and area interpolation to reduce image size, retraining the model at different resolutions.

Figure 5 [Figure 5: see original paper] shows the impact of different resolutions as inputs on regression and classification results. Within the 8–128 range, as resolution increases, the model' s accuracy improves to some extent, suggesting that higher resolution images can provide more detailed information within a certain range, ultimately aiding the model in better learning and recognizing different features. However, when image resolution is scaled up to 224 × 224 pixels, the accuracy of both classification and regression decreases. This may be due to the bilinear interpolation method introducing excessive blur or artifacts while enlarging the image, leading to reduced accuracy. In summary, to balance accuracy and computational efficiency, a resolution of 64 × 64 pixels is considered the best choice for this study.

4.4. The Impact of Excluding Parallax and Proper Motion Data

In this section, we keep the photometric images as fixed input and compare the impact of using different inputs on model performance. For different inputs, we retrained using MESCR, with results shown in Table 2 .

For classification results, photometric images provide unique color and brightness characteristics of different stars. Therefore, the model can achieve good classification results even when using only photometric images as input. After introducing parallax, the F1 scores for A, F, and G-type stars all improved because parallax indirectly infers the absolute brightness of stars and determines their actual distances, thereby improving classification accuracy. As for proper motion, although it introduces dynamic information about stellar movement, the improvement in accuracy is smaller compared to parallax. When photometric images, parallax, and proper motion are used together as input, the model shows optimal performance in overall classification accuracy, indicating that integrating data from multiple modalities can significantly enhance predictive accuracy.

For stellar radius estimation, our experimental results show that different inputs have a slight impact on MAE and σ. This may be due to the limited information provided by additional parallax and proper motion for predicting stellar radii, and the model has certain stability in radius prediction. Considering that radius can be derived from temperature using the Stefan–Boltzmann law, we used the Teff_{gspphot} provided by Gaia DR3 as an input to check whether temperature can constrain the radius. However, results showed that radius estimation accuracy did not improve significantly after including temperature. In future work, we will continue to explore other factors that may affect radius

estimation accuracy and further optimize the model.

5. Application To Spectral-Free Data

In previous work, [**?**] used a random forest machine learning model to classify target sources into three categories: stars, galaxies, and quasars, ultimately creating a catalog containing 111,395,468 objects (hereinafter referred to as the Clarke Catalog). Of these, 58,840,082 are stars, and none had corresponding spectra in SDSS. [**?**] further cross-matched the Clarke Catalog with LAMOST, using LAMOST's classification results as reference values. They found that when class_{prob}_{star} (the probability of being predicted as a star) exceeded 0.8, the accuracy was 0.99. To ensure data reliability, we further filtered the Clarke Catalog to include only stellar samples with class_{prob}_{star} $>$ 0.8. We then cross-matched these with Gaia DR3 using TOPCAT ([**?**]) by R.A. and decl. within a radius of three arcseconds to obtain corresponding parallax and proper motion. The resulting catalog contains 50,871,534 stars without spectra but with associated parallax and proper motion. We applied MESCR to this catalog, with columns shown in Table A2. It is important to note that none of the stars processed by MESCR exceed the limiting magnitude. The entire catalog is available online via the China-VO PaperData repository.

6. Conclusion

This paper introduced an ensemble learning scheme named MESCR based on MMSCNet. MESCR integrates multimodal data to simultaneously perform star classification and radius estimation. To obtain the final training samples, we first performed cross-matching to acquire the necessary data, then cropped the original images to reduce irrelevant information. Subsequently, we weighted sampled the dataset to obtain six different datasets and trained six sub-models separately, finally integrating the predictions of all sub-models to produce the final output. Our model achieved a classification accuracy of 96.1% on the test set, with the MAE for stellar radius at 0.084 dex and σ at 0.149 R . Through uncertainty analysis, we further proved that our MESCR solution has good accuracy and robustness.

We tested the impact of different band combinations and resolutions on model performance, finding that specific band combinations and appropriate image resolutions can significantly optimize model performance. Additionally, we quantitatively assessed the impact of using data from different modalities as inputs on model accuracy. The results indicate that integrating various modalities of data can enhance the model' s ability to distinguish between different types of stars. Ultimately, we applied MESCR to 50,871,534 stars without spectra and released a new catalog.

In our future work, we will further improve model precision and explore more survey data, while also enhancing the accuracy of our catalog to provide greater reference value for astronomers and data analysis researchers.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (12261141689, 12273075, and 12373108), the National Key R&D Program of China No. 2019YFA0405502, and the science research grants from the China Manned Space Project with No. CMS-CSST-2021-B05.

Data Availability

The complete code of this study and its instructions for use will be published at https://github.com/yuu250/MESCR after this paper is accepted and finalized, for reference by astronomers and data processing personnel.

Appendix: Supplementary Figures and Tables

This appendix contains supplementary images and tables discussed and analyzed in the article. These materials provide detailed visual evidence and data support to enhance the reader' s understanding.

Figure A1. An example of synthesizing photometric images. The first row represents RGB images synthesized from the gri bands, while the second row represents RGB images synthesized from the urz bands. The urz band images exhibit more noise compared to the gri band images.

Figure A2. Image cropping schematic. The size of the image before cropping is 64×64. The left image shows the initial Region of Interest (ROI) with a size of 6×6. The right image shows the final cropped target ROI, with most ROIs sized between 10×10 and 20×20.

Figure A3. Randomly selected M-type star showing the effect of data augmentation. Titles display the corresponding data augmentation methods. In this study, both gri and urz images undergo the same data augmentation methods. The first row shows the effects of data augmentation on gri images, while the second row displays the effects on urz images.

Figure A4. The six subfigures represent the distribution of stellar spectral classes for six datasets after weighted sampling and division into training and validation sets.

Figure A5. After weighted sampling and splitting into training, validation, and test sets for six datasets, the corresponding distribution of proper motion, parallax, and radius_{flame}. Note that the six datasets use the same test set, so the distribution of the test set remains consistent.

Figure A6. The loss curves for each model during the training process on both training and validation sets. The left graph represents changes in loss values for the training set, while the right graph shows changes for the validation set. Loss curves for different models are distinguished using different colors.

Table A1. Evaluation results on the test set after retraining the model with different band combinations. Note: Bold indicates the best performance.

Band Combination	Accuracy	F1-score
gri+urz	94.9%	95.6%
giu+grz	94.7%	95.2%
giu+riz	95.3%	95.3%
giu+ruz	95.7%	95.7%
giz+riu	95.2%	95.5%
giz+ruz	95.5%	95.5%
gri+guz	95.1%	95.1%
gri+iuz	95.7%	95.5%
gri+ruz	94.3%	94.5%
gru+giz	94.5%	95.6%
gru+iuz	95.2%	95.2%
gru+riz	94.9%	94.9%
guz+riu	94.8%	94.8%
guz+riz	92.7%	95.2%
iuz+grz	95.2%	95.1%
riu+grz	94.9%	95.5%
gri+uz	94.6%	95.2%
giu+rz	95.3%	95.3%
giz+ru	95.7%	95.6%
gru+iz	95.1%	95.4%
grz+iu	95.4%	95.4%
guz+ri	95.1%	95.1%
iuz+gr	94.2%	94.3%
riu+gz	94.9%	94.9%
riz+gu	94.7%	94.6%
ruz+gi	93.3%	95.2%
iuz+gr	95.2%	95.1%

Table A2. Description of our catalog.

Column	Column Name	Description
objid	Unique SDSS object identifier	
ra	J2000 R.A. (r-band)	
decl	J2000 decl. (r-band)	
psfMag_u (g, r, i, z)	SDSS PSF magnitude in the u(g, r, i, z) band	
spectral_{class}	Class of model predictions	

Column	Column Name	Description
prob_O(B, A, F, G, K, M)	Probability of prediction as an O(B, A, F, G, K, M)-type star	
radius	Radius of model predictions	
radius_{err}	The uncertainty provided by the model when predicting the radius	

Note: Figure translations are in progress. See original paper for figures.