



Research Article

Estimating stellar parameters and identifying very metal-poor stars for low-resolution spectra ($R \sim 200$)

Tianmin Wu^{1,2,3}, Yude Bu³, Jianhang Xie³, Junchao Liang³, Wei Liu³, Zhenping Yi⁴, Xiaoming Kong⁴ and Meng Liu⁴

¹CAS Key Laboratory of Optical Astronomy, National Astronomical Observatories, Chinese Academy of Sciences, Beijing 100101, People's Republic of China, ²School of Astronomy and Space Science, University of Chinese Academy of Sciences, Beijing 100049, People's Republic of China, ³School of Mathematics and Statistics, Shandong University, Weihai, Shandong 264209, People's Republic of China and ⁴School of Mechanical, Electrical & Information Engineering, Shandong University, Weihai, Shandong 264209, People's Republic of China

Abstract

Very metal-poor (VMP, $[\text{Fe}/\text{H}] < -2.0$) stars serve as invaluable repositories of insights into the nature and evolution of the first-generation stars formed in the early galaxy. The upcoming China Space Station Telescope (CSST) will provide us with a large amount of spectral data that may contain plenty of VMP stars, and thus it is crucial to determine the stellar atmospheric parameters (T_{eff} , $\log g$, and $[\text{Fe}/\text{H}]$) for low-resolution spectra similar to the CSST spectra ($R \sim 200$). This study introduces a novel two-dimensional Convolutional Neural Network (CNN) model, comprised of three convolutional layers and two fully connected layers. The model's proficiency is assessed in estimating stellar parameters, particularly metallicity, from low-resolution spectra ($R \sim 200$), with a specific focus on enhancing the search for VMP stars within the CSST spectral data. We mainly use 10 008 spectra of VMP stars from LAMOST DR3, and 16 638 spectra of non-VMP stars ($[\text{Fe}/\text{H}] > -2.0$) from LAMOST DR8 for the experiments and apply random forest and support vector machine methods to make comparisons. The resolution of all spectra is reduced to $R \sim 200$ to match the resolution of the CSST, followed by pre-processing and transformation into two-dimensional spectra for input into the CNN model. The validation and practicality of this model are also tested on the MARCS synthetic spectra. The results show that using the CNN model constructed in this paper, we obtain Mean Absolute Error (MAE) values of 99.40 K for T_{eff} , 0.22 dex for $\log g$, 0.14 dex for $[\text{Fe}/\text{H}]$, and 0.26 dex for $[\text{C}/\text{Fe}]$ on the test set. Besides, the CNN model can efficiently identify VMP stars with a precision rate of 94.77%, a recall rate of 93.73%, and an accuracy of 95.70%. This paper powerfully demonstrates the effectiveness of the proposed CNN model in estimating stellar parameters for low-resolution spectra ($R \sim 200$) and recognizing VMP stars that are of interest for stellar population and galactic evolution work.

Keywords: Convolutional Neural Network; very metal-poor stars; LAMOST; low-resolution spectra; CSST

(Received 25 November 2022; revised 3 November 2023; accepted 7 November 2023)

1. Introduction

Very metal-poor (VMP, $[\text{Fe}/\text{H}] < -2$) stars are important relics of the galactic formation history since it offers rich information on the chemical and physical conditions of the first-generation stars in the universe. The Li abundance in VMP stars can provide estimates of the baryon-to-photon ratio, helping us to better refine the galaxy formation model (Beers and Christlieb 2005; Frebel and Norris 2015; Frebel 2018).

Owing to large-scale sky survey projects, researchers have so far discovered a large number of VMP stars using photometry and spectroscopic methods. One of the earliest VMP search programmes was HK survey (Rhee, Beers, & Irwin 2001), discovering approximately 2 000 cooler VMP stars by combining Artificial Neural networks and 2MASS JHK photometry, with a detection efficiency of between 60% and 70%. Christlieb et al. (2008) used CCD photometry to calibrate B-V colours in the Hamburg/ESO Survey (HES) spectra and found 10 times more VMP stars than

the HK survey, with a rejection rate of 97% for stars with $[\text{Fe}/\text{H}] > -2$. The Pristine Inner Galaxy Survey (PIGS) photometry (Arentsen et al. 2020) explored 1 300 VMP stars in the inner Galaxy with an efficiency exceeding 80%. For the spectroscopic survey, APOGEE (García Pérez et al. 2013) also found a small number of VMP stars in the central part of the Milky Way. Besides, Matijević et al. (2017) determined the metallicity of data from RAVE DR5 more reliably than the pipeline, with an accuracy of 0.2 dex, and found hundreds of VMP stars. Da Costa et al. (2019) discovered nearly 2 500 VMP stars from SkyMapper DR1.1 with the help of the follow-up low-resolution ($R \sim 3 000$) spectroscopic research. Other surveys such as LAMOST (Li, Tan, & Zhao 2018) and Southern African Large Telescope (SALT, Rasmussen et al. 2020) also enlarge the sample pool of VMP stars.

Metal-poor stars tend to contain higher than average levels of carbon. If the carbon abundance ($[\text{C}/\text{Fe}]$) of the metal-poor ($[\text{Fe}/\text{H}] < -1.0$) stars is larger than +1.0, it is called the Carbon Enhanced Metal-poor (CEMP) stars (Beers and Christlieb 2005). This threshold for classifying CEMP stars has been updated to $[\text{C}/\text{Fe}] > +0.7$ (Aoki et al. 2007). Measuring the carbon enhancement of metal-poor stars discovered from large-scale surveys is conducive to deriving CEMP stars, which are of vital importance for understanding the relationship between astrophysical

Corresponding author: Yude Bu; Email: buyude@sdu.edu.cn

Cite this article: Wu T, Bu Y, Xie J, Liang J, Liu W, Yi Z, Kong X and Liu M. (2024) Estimating stellar parameters and identifying very metal-poor stars for low-resolution spectra ($R \sim 200$). *Publications of the Astronomical Society of Australia* 41, e002, 1–12. <https://doi.org/10.1017/pasa.2023.59>

s-process and carbon enhancement (Marsteller et al. 2005) and the nature of first-generation stars (Beers 2011). Frebel et al. (2006) proved that there is clear evidence that the proportion of carbon enrichment in metal-poor stars increases with decreasing metallicity. Lucatello et al. (2006) first analysed 94 VMP stars obtained by the Hamburg/ESO R-process Enhanced Star survey (HERES, Barklem et al. 2005) and found $21 \pm 2\%$ of VMP stars with $[C/Fe]$ abundances above $+1.0$, which can be classified as CEMP stars. Lee et al. (2013) proposed a novel technique to obtain the fractions of CEMP stars in metal-poor stars from a large sample of SDSS/SEGUE low-resolution ($R \sim 2\,000$) spectra with precision over 0.35 dex. Placco et al. (2014) improved the frequency of CEMP stars in metal-poor stars and derived that 20% of VMP stars have $[C/Fe]$ abundance greater than $+0.7$ utilizing the most extensive high-resolution samples from a variety of literature at that time.

In order to identify VMP and CEMP stars, the metallicity ($[Fe/H]$) and $[C/Fe]$ abundance should first be determined. Many methods have been proposed to extract stellar atmospheric parameters (effective temperature T_{eff} , surface gravity $\log g$, and metallicity $[Fe/H]$) and chemical abundances from large amounts of spectral or photometric data. The spectral template fitting method is the most widely used method and has high reliability. Lee et al. (2008) proposed the SEGUE Stellar Parameter Pipeline (SSPP) using the line index method and χ^2 minimum spectral fitting method and tested its performance on SDSS-I and SDSS-II/SEGUE medium-resolution spectra. Koleva et al. (2009) presented a full-spectrum fitting package ULYSS by fitting the minimum observed spectra and model. Blanco-Cuaresma et al. (2014) created iSpec using synthetic spectral fitting and equivalent width methods based on the SPECTRUM code and tested its effectiveness in the *Gaia* stellar spectral library. Wu et al. (2014) developed LAMOST stellar parameter pipeline (LASP) using the correlation function interpolation (CFI) and ULYSS methods to automatically derive stellar parameters and radial velocity (V_r) for late A, FGK-type stars.

With the development of machine learning and the maturity of artificial intelligence technology, more and more deep learning methods are applied to stellar parameter measurement. Ness et al. (2015) developed The Cannon, a data-driven model that does not rely on physical models, which allows us to derive stellar labels from low signal-to-noise (S/N) spectra. StarNet (Fabbro et al. 2018) and SPCANet (Wang et al. 2020) both employed Convolutional Neural Network (CNN), which is an approach capable of automatically learning data features without the need for manual feature design (Krizhevsky, Sutskever, & Hinton 2012; Szegedy et al. 2015). This can save time and costs, making it more suitable for application to large astronomical datasets. Leung and Bovy (2019) used the Artificial Neural Network (ANN), CNN, and Bayesian dropout variational inference to successfully obtain 18 element abundances of APOGEE high-resolution spectra. Ting et al. (2019) presented Payne by combining the neural network spectral interpolating method and physical models, which can determine a variety of stellar labels simultaneously.

These deep learning methods are a good solution to the problem of slow speed of traditional template matching methods, but most of them are still based on high-resolution spectra or spectra with $R \sim 2\,000$ down to 1 000. The Chinese Space Station Telescope (CSST) (Zhan 2021) to be emitted in the future will acquire slitless spectra (Yuan, Deng, & Sun 2021), i.e. on the focal plane, all sources are able to disperse uniformly along the

dispersion direction, which allows us to acquire the full spectrum of the entire field of view. However, the resulting mixing of the dispersion terms leads to contamination, which makes data processing more difficult and also further reduces the resolution ($R \sim 200$) and signal-to-noise ratio of the spectra. Moreover, the low metal abundance and weak spectral line features of VMP stars make most stellar parameter estimation methods inefficient for VMP stars. Both of these reasons increase the difficulty of using CSST spectral data to estimate stellar labels and identify VMP stars. So far, several studies have already shown the feasibility of estimating stellar parameters from very low-resolution spectra by using *Gaia* BP/RP spectra with a resolution of about 50 (Gavel et al. 2022; Witten et al. 2022). In this paper, we construct a two-dimensional CNN model composed of three convolutional layers and two fully connected layers. It can be an important supplement to the above methods and will be of great help for future studies of CSST. We use the spectral data obtained from LAMOST and reduce its resolution to $R \sim 200$ to validate our model. MARCS synthetic spectra and other machine learning methods are also used to test whether our model has higher accuracy.

The paper consists of five parts. The data selection and data pre-processing are introduced in Section 2. Then we describe the principles of CNN models and the structure of our model in Section 3. Section 4 presents the experiments and results. Section 5 discusses the comparison between CNN models and other machine learning algorithms. Brief conclusions can be seen in Section 6.

2. Data

The data used in the experiments in this paper are obtained from the LAMOST database and MARCS synthetic spectra. This section briefly introduces the data sources and selection process, as well as the pre-processing of the data.

2.1. LAMOST

2.1.1. Data introduction

In 2009, the Large Sky Area Multi-Object Fiber Spectroscopic Telescope (LAMOST) was successfully completed in China. As an exemplary representative among spectroscopic survey telescopes, LAMOST utilizes advanced thin mirror active optics and spliced mirror active optics technology, ingeniously combining a large field of view with a substantial aperture. This achievement has enabled LAMOST to provide an unparalleled understanding of the Milky Way and has significantly advanced the development of large-aperture astronomical telescopes in China (Li et al. 2022). Thanks to LAMOST's enhanced observing capabilities and its ability to acquire large multi-fibre samples, the observation of medium or low-resolution spectra ($R \sim 1\,000$ or $2\,000$) from LAMOST has made substantial contributions to the search for VMP stars (Wu et al. 2010; Li et al. 2015, 2018; Wang et al. 2022). To date, LAMOST has publicly released its eighth version of data, comprising an impressive collection of 10 633 515 low-resolution ($R \sim 1\,800$) spectra, covering 10 336 752 stars, 224 702 galaxies, and 72 061 quasars. The LAMOST stellar parameter pipeline (LASP) automatically derives the basic stellar atmospheric parameters for A, F, G, and K types of stars (Wu et al. 2014), while the determination of M-type stars is carried out by LASP-M (Du et al. 2021).

2.1.2. Data selection

In order to compare the ability of the model to estimate stellar parameters for VMP stellar spectra and non-VMP stellar spectra, and the accuracy of identifying VMP stars, we construct a dataset including VMP stars and non-VMP stars ($[\text{Fe}/\text{H}] > -2.0$) as positive and negative samples. To ensure the reliability of the stellar parameters, we combine two stellar parameter sources. For the VMP stellar sample, we choose the catalog containing 10 008 VMP stars obtained from LAMOST DR3 by Li et al. (2018) since it still provides researchers with the largest pool of bright and accurate VMP candidates to date. The stellar atmospheric parameters of these VMP stars are determined by line indices and by comparison with a grid of synthetic spectra, with metallicity ranging from -4.5 dex to -2.0 dex. For the negative sample dataset, we randomly select data with $[\text{Fe}/\text{H}] > -2.0$ from the LAMOST DR8 dataset that is recently made public to the world, where the stellar atmospheric parameters are determined by LASP (Wu et al. 2014). To ensure that the accuracy of the stellar parameters of this dataset is comparable to that of the VMP stars dataset, we obtain 16 638 non-VMP stars with minor uncertainties and a signal-to-noise ratio larger than 10 at the g-band. In total, we construct a dataset including 26 646 stars. The resolution of all the spectral data is $R \sim 1800$. The parameters of these stars range from $3824.88 \text{ K} < T_{\text{eff}} < 8866.15 \text{ K}$, $0.213 \text{ dex} < \log g < 4.897 \text{ dex}$, and $-4.55 \text{ dex} < [\text{Fe}/\text{H}] < 0.699 \text{ dex}$. The errors of the parameters range from $0 \text{ K} < \sigma(T_{\text{eff}}) < 399 \text{ K}$, $0 \text{ dex} < \sigma(\log g) < 0.94 \text{ dex}$, and $0 \text{ dex} < \sigma([\text{Fe}/\text{H}]) < 0.4 \text{ dex}$. For subsequent model training and testing, we divide this dataset into a training set and a test set in the ratio of 7:3, including 18 652 and 7 994 stars, respectively. Fig. 1 depicts more clearly the distribution of the parameters in the training and test sets.

To estimate the carbon abundance and identify CEMP stars in VMP stars, we need a dataset of VMP stars containing $[\text{C}/\text{Fe}]$ labels. Since the VMP stars catalog mentioned above does not contain $[\text{C}/\text{Fe}]$ values, Yuan et al. (2020) cross-matched this catalog with *Gaia* DR2 and obtained a modified catalog including 9 690 stars with parameters derived by SEGUE Stellar Parameter Pipeline (SSPP, Lee et al. 2008). We cross-match the VMP stars catalog (Li et al. 2018) with the modified catalog (Yuan et al. 2020) and find 8 117 of 10 008 VMP stars with $[\text{C}/\text{Fe}]$ labels. The range of the $[\text{C}/\text{Fe}]$ labels is $-2.018 \text{ dex} < [\text{C}/\text{Fe}] < 4.803 \text{ dex}$. For the aim of model training and testing, we divide the dataset into a training set and a test set in the ratio of 7:3, containing 5 681 and 2 436 stars, respectively.

2.1.3. Data pre-processing

To obtain the data needed for the experiment, we preprocess the spectral data as follows:

(1) Resolution reduction:

Reduce the resolution of the spectra from $R \sim 1800$ to $R \sim 200$ to simulate the low-resolution spectra acquired by CSST. The *Coronagraph* library provided in Python with the `noise_routines.construct_lam()` and `downbin_spec()` functions can bring the data down to the resolution we need and output the degraded flux.

Fig. 2 shows a comparison of the spectra with resolution $R \sim 1800$ and $R \sim 200$. Fewer features in the lower resolution spectra make the stellar parameter estimation more difficult.

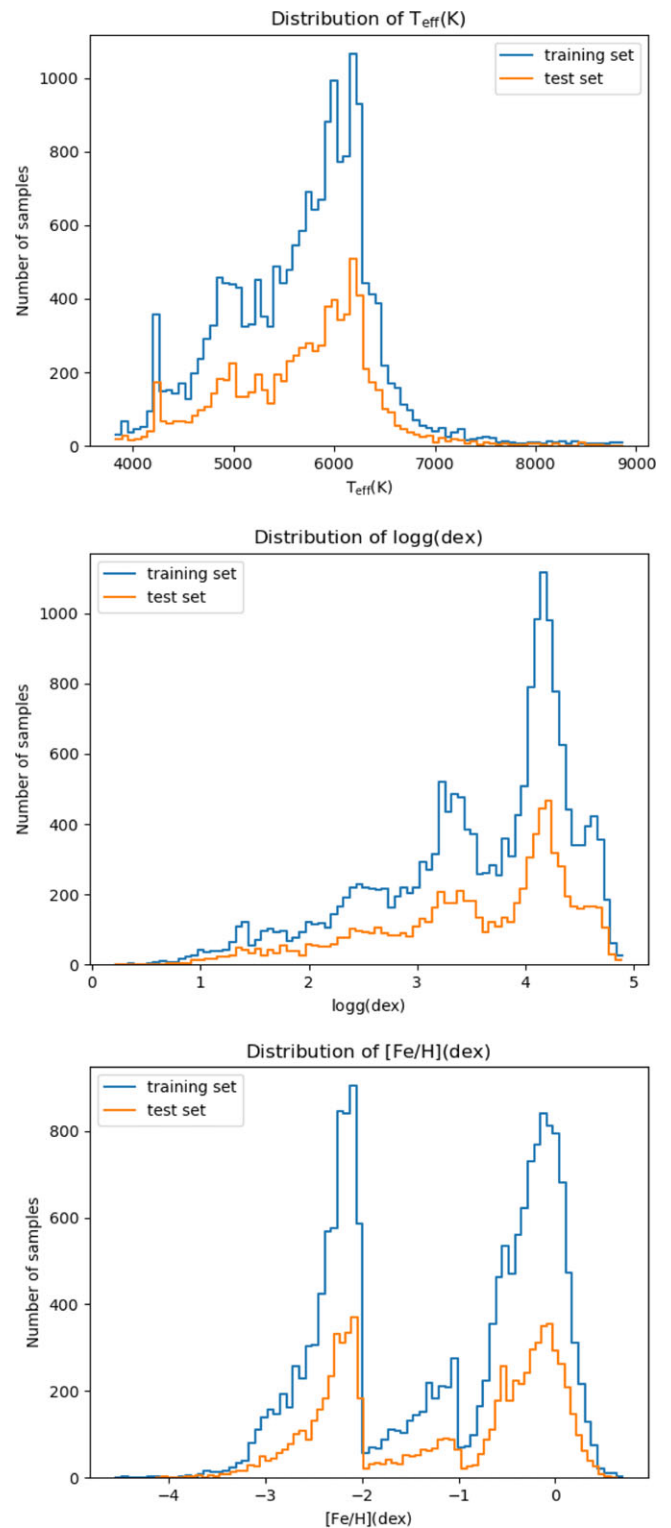


Figure 1. The distribution of T_{eff} (top panel), $\log g$ (middle panel), and $[\text{Fe}/\text{H}]$ (bottom panel) in the LAMOST training set and test set. The division of the training and test sets ensures the consistency of the data distribution.

(2) Interpolation:

Interpolate the flux data to the range of 4000\AA to 8095\AA , ensuring that the data is sampled at the same wavelengths.

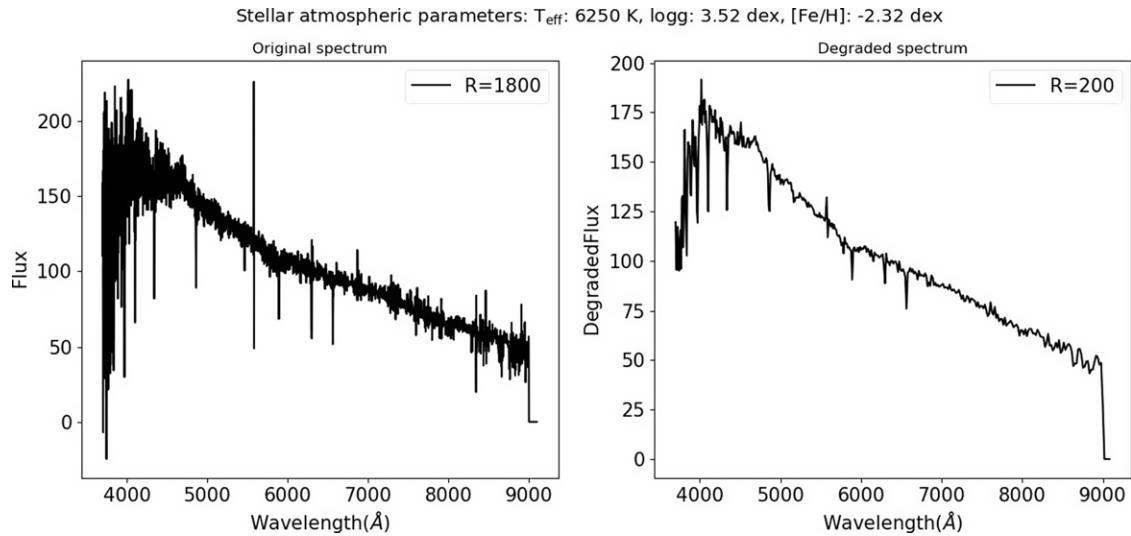


Figure 2. The plots of spectra with resolution 1 800 (left) and 200 (right). Spectra with lower resolution have fewer characteristic points.

This results in a consistent flux range for all spectra and reduces the effect of noise at both ends of the spectrum.

(3) Normalization:

The flux values are then normalized by a linear function (Min-Max scaling), which can achieve equal scaling of the original data to convert the flux to the range of [0, 1], as follows.

$$Flux_{norm} = \frac{Flux - Flux_{min}}{Flux_{max} - Flux_{min}}, \quad (1)$$

Finally, we obtain spectral data with 410 feature points. Section 4.1 shows the results of estimating stellar labels and identifying VMP stars in the LAMOST dataset using our CNN model.

2.2. MARCS dataset

To further test the validity of the CNN model for estimating stellar parameters, we use the MARCS synthetic spectra (Gustafsson et al. 2008). It is a grid of about 10^4 model atmospheres with nearly 52 000 stellar spectra containing F, G, and K types of stars. This grid of one-dimensional LTE model atmospheres can be combined with atomic and molecular spectral line data and software to generate stellar spectra, which has been widely used in a variety of studies (Roederer et al. 2014; Lu et al. 2018; Reggiani et al. 2019; VandenBerg et al. 2021; Salsi et al. 2022).

We select 9 644 data from MARCS for the experiment. The range of the stellar parameters is $2\,500\text{ K} < T_{\text{eff}} < 8\,000\text{ K}$, $-0.5\text{ dex} < \log g < 5.5\text{ dex}$, $-5\text{ dex} < [\text{Fe}/\text{H}] < -1\text{ dex}$, and the step size of the parameters is 2 500 K for T_{eff} , 0.5 dex for $\log g$, and 0.25 dex for $[\text{Fe}/\text{H}]$. We perform the same interpolation and normalization operations on the data and obtain spectral data with 746 feature points. The dataset is divided into a training set and a test in the ratio of 7:3, including 6 750 and 2 894 stars, respectively. Section 4.3 shows the results of estimating stellar parameters of the MARCS synthetic spectra using the CNN model.

3. Methodology

3.1. Introduction to the convolutional neural network (CNN)

In this paper, we construct a CNN model and test its performance in estimating stellar labels for low-resolution ($R \sim 200$) spectra and identifying VMP stars. This can enable us to better handle low-resolution spectra of CSST in the future and search for VMP stars.

The concept of deep learning originated from the study of artificial neural networks and was proposed by Hinton (2008). CNN (Lecun et al. 1998) is a typical supervised model of deep learning that has been widely applied to various fields in recent years. It is the first actual multilayer structure learning algorithm that uses spatial relative relationships to reduce the number of parameters to improve training performance. It adds a feature part to the original multilayer neural network; i.e. a convolutional layer and a pooling layer (dimensionality reduction layer) are added before the fully connected layer, and the network selects the features itself. CNN is a deep feedforward neural network that is widely used for supervised learning problems in image processing and natural language processing, such as computer vision (Krizhevsky et al. 2012), semantic segmentation (Ronneberger, Fischer, & Brox 2015), object recognition (Redmon et al. 2016), etc.

While the CNN model can directly process one-dimensional sequences such as spectral data, most current CNN models demonstrate their power in processing two-dimensional image data. In the LAMOST dataset, the length of the 1D spectral data (1×410) we obtain is long, and collapsing it into 2D data (21×21) to simulate image data can perform cross-correlation operations and extract spectral features better and faster, thus speed up the network learning speed and parameter optimization efficiency. For MARCS spectral data (1×746), we also collapse them into a two-dimensional matrix (28×28) as the input to the CNN model. The following is a brief description of the structure and principles of the 2D CNN.

A complete CNN model must include convolutional layers, non-linear activation functions, pooling layers, and fully connected layers.

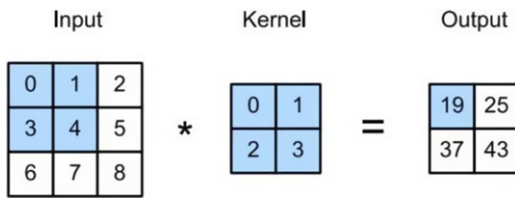


Figure 3. Convolution process.

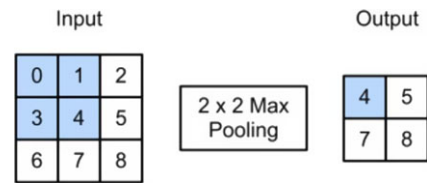


Figure 4. Max-pooling process.

(1) Convolutional layer:

The convolutional layer is the core layer for establishing the CNN model, which can act as a filter and reduce the number of parameters. The discrete form of two-dimensional convolution is used.

Given a figure $X \in \mathbf{R}^{M \times N}$ and a convolutional kernel $W \in \mathbf{R}^{U \times V}$. In general, $U < M, V < N$. The convolution between them can be denoted as

$$Y = W * X, \tag{2}$$

$$y_{ij} = \sum_{u=1}^U \sum_{v=1}^V w_{uv} x_{i-u+1, j-v+1}. \tag{3}$$

An example of the convolution process is shown in Fig. 3. A two-dimensional input array (3 × 3) performs a mutual correlation operation with a two-dimensional convolutional kernel array (2 × 2), resulting in a two-dimensional array (2 × 2). The convolution kernel slides over the input array from left to right and top to bottom.

The convolutional layer extracts features from local regions, and different convolutional kernels are equivalent to various feature extractors. Based on the standard definition of convolution, strides and zero padding of the convolutional kernel can be introduced to increase the diversity of convolution. Strides refer to the number of steps each convolutional kernel moves when performing a convolutional operation. Set stride= k , which means convolving k rows and k columns from left to right and from top to bottom. Zero padding represents adding zeros to the outer side of the image. Setting padding= d means supplementing d layers of zeros around the input vector. Zero padding allows us to obtain more detailed feature information and control the network structure.

After the convolutional layer, an activation function is usually added as a non-linear factor, which can deal with problems that cannot be solved by linear models and enhance the ability of the network to interpret the model. A commonly used activation function is ReLU (Equation (4)). It has a small computational effort, and can effectively alleviate the gradient disappearance and gradient explosion problem because its derivative is maintained at 1.

$$f(x) = \max(0, x). \tag{4}$$

(2) Pooling layer:

The pooling layer (subsampling layer) is designed to reduce the number of features in the network, thereby reducing the number of parameters and avoiding overfitting. Local translation invariance is an essential property

of pooling layers, which indicates that pooling is approximately invariant in its representation of the input when a small number of translations are performed on the input. Max-pooling is the most commonly used which means extracting the maximum value within the neighbourhood (see Fig. 4). It can reduce the offset of the estimated mean value caused by the parameter error of the convolutional layer, and retain more texture information.

(3) Fully connected layer:

The purpose of the fully connected layer is to connect the results of the last pooling layer to the output nodes and map the feature representation learned by the network to the label space of the samples. It acts as the ‘classifier’ of the network. It is important to note that when encoding the model, the last pooling layer needs to be flattened to a one-dimensional vector before connecting to the fully connected layer.

CNN extracts high-level semantic information from the input data gradually through a series of operations in these layers, formalizes the target task as an objective function, and generates the predicted values. This process is called Forward Propagation. The parameter optimization in the model is performed by Back Propagation. By calculating the loss between the predicted values and the true values, the CNN feeds the loss from the previous layer to the next layer, calculates the gradient of the loss on the parameters of each layer, and updates the corresponding parameters. Once all the parameters have completed one round of updates, the feedforward operation is performed again. This process is repeated iteratively until the model converges.

3.2. The method of preventing overfitting

When the number of parameters in a model is large compared to the available training samples, overfitting can occur. This is manifested by high prediction accuracy on the training set and a significant decrease in accuracy on the test set.

The advent of dropout (Hinton et al. 2012) has greatly alleviated this problem. During training, each neuron is retained with probability p and stops working with probability $1 - p$, and a different set of neurons is retained for each forward propagation. This approach reduces the reliance of the model on certain local features and has better generalization performance.

Another way is to add Batch Normalization (BN, Ioffe and Szegedy 2015) layers after the convolutional and fully connected layers. During the training process of neural networks, parameter changes can lead to unstable distribution of activation values, which hinders the ability of subsequent layers to learn useful features. In addition to normalizing the input data of each layer and promoting a stable distribution of activation values, the BN layer can accelerate the convergence thus preventing model overfitting.

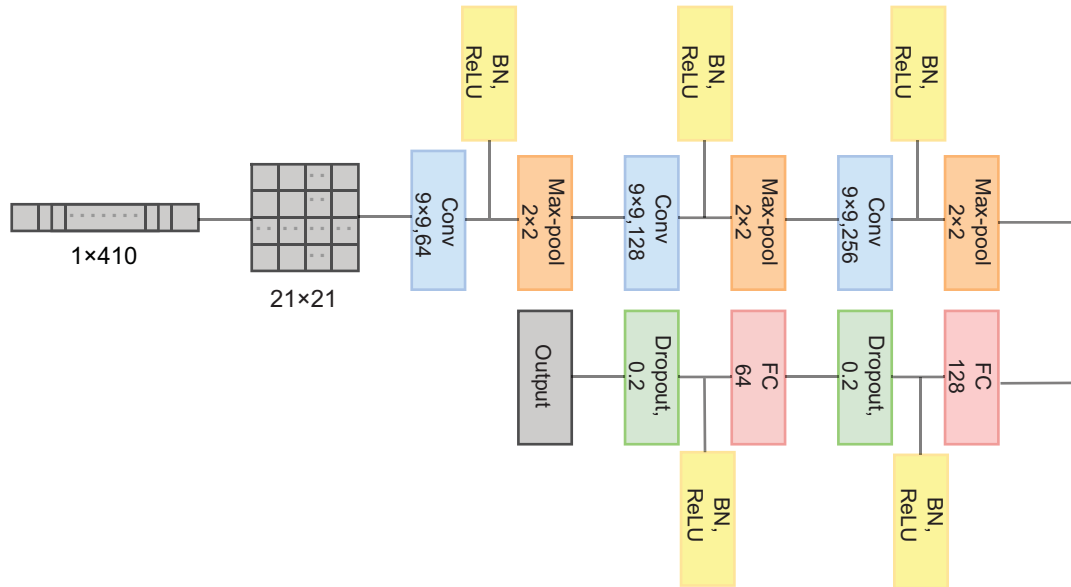


Figure 5. The structure of the proposed CNN model. The 1D spectral data is transformed into a 2D image-like matrix and is then input into the neural network consisting of three convolutional layers, three max-pooling layers, and two fully connected layers to extract the stellar parameters.

3.3. The structure of the proposed CNN model

After parameter tuning of the model, the final CNN model we construct is shown in Fig. 5. The convolutional layers filter the processed 2D spectra using a filter of size 9×9 , and the input is filled with zero space on the boundary so that the size of the output layer of the convolution is equal to the size of the input layer. The 9×9 convolution kernel can acquire a larger field of perception and therefore can capture more characteristics. The kernels of each convolutional layer are 64, 128, and 256, respectively. Each convolutional layer is followed by a max-pooling layer of size 2×2 with a step size set to two. Afterwards, two fully connected layers with 128 and 64 channels are added to combine the features previously extracted by the model. To prevent overfitting, a dropout layer can be set after each fully connected layer with a value of 0.2 to avoid the over-regularization of the model. BN and ReLU activation function layers are added between each layer to reduce overfitting and enhance the expressiveness of the model. The final output layer is the predicted values derived from the model.

3.4. Experimental procedure

To test whether our model can better estimate stellar parameters and identify VMP stars, we conduct experiments using Python 3.8 on NVIDIA GeForce RTX GPU. As mentioned in Section 2, both the LAMOST dataset and the MARCS dataset are divided into training and testing sets in the ratio of 7:3 to train the model and test its effectiveness. The training and testing process is performed on T_{eff} , $\log g$, $[\text{Fe}/\text{H}]$, and $[\text{C}/\text{Fe}]$ respectively.

We set a total of 1 000 epochs for model training, and L1loss (MAE, Equation (5)) is used as the loss function. The training process is performed in batches, with the size of each batch set to 128. The Adam algorithm (Kingma and Ba 2014), with an initial learning rate set to 0.001, is chosen to be the optimizer, which is an extension of the stochastic gradient descent method. Adam optimizer can speed up convergence by adapting the learning rate,

making it well-suited for deep learning problems. An early stopping mechanism is set when the loss function no longer decreases beyond 250 epochs, which can effectively prevent the model from overfitting.

3.4.1. Evaluation metrics

We use three main evaluation metrics to test the effectiveness of the model in estimating stellar parameters, Mean Absolute Error (MAE, Equation (5)), Standard Deviation (STD, Equation (6)), and R squared (R^2 , Equation (7)).

Suppose that N is the number of samples contained in the test set, y denotes the true values, and \hat{y} denotes the predicted values derived by the proposed model. Let e_i be $y_i - \hat{y}_i$, and \bar{e}_i be the average value of e_i .

- (1) Mean absolute error (MAE): MAE is a loss function used in regression models, which can express the fitting ability of the model more intuitively.

$$MAE(y, \hat{y}) = \frac{1}{N} \sum_{i=0}^{N-1} |y_i - \hat{y}_i|. \quad (5)$$

- (2) Standard deviation (STD): Standard deviation reflects the degree of dispersion of a dataset.

$$STD(y, \hat{y}) = \sqrt{\frac{1}{N} \sum_{i=1}^N (e_i - \bar{e}_i)^2}. \quad (6)$$

- (3) R squared (R^2): The numerator is the error between the predicted value and the true value, and the denominator is understood as the dispersion of the true value. The division of the two can eliminate the effect of the dispersion of the

Table 1. The prediction results of the three fundamental atmospheric parameters on the test set including 7 994 stars using the proposed CNN model.

Parameter	MAE	STD	R^2
T_{eff} (K)	99.40	183.33	0.93
log g(dex)	0.22	0.35	0.84
[Fe/H](dex)	0.14	0.26	0.94

true value. The closer R^2 is to 1, the better the model fits the data.

$$R^2(y, \hat{y}) = 1 - \frac{\sum (y - \hat{y})^2}{\sum (y - \bar{y})^2}. \tag{7}$$

Three other metrics can be used to test the effectiveness of the model in classifying stars, precision (Equation (8)), recall (Equation (9)), and accuracy rate (Equation (10)).

- (1) True Positive (TP): VMP stars predicted as VMP stars.
- (2) True Negative (TN): Common stars predicted as common stars.
- (3) False Positive (FP): Common stars predicted as VMP stars.
- (4) False Negative (FN): VMP stars predicted as common stars.

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{8}$$

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{9}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}. \tag{10}$$

4. Results

In this section, we show the parameter estimation and classification results in the dataset described in Section 2 using the proposed CNN model.

4.1. Estimating stellar parameters and identifying VMP stars using the LAMOST dataset

We start our experiments with a total of 26 646 stars, including both VMP and non-VMP stars. The prediction results obtained on the test set are displayed in Table 1. For T_{eff} , MAE=99.40 K, STD=183.33 K, $R^2 = 0.93$; for log g, MAE=0.22 dex, STD=0.35 dex, $R^2 = 0.84$; and for [Fe/H], MAE=0.14 dex, STD=0.26 dex, $R^2 = 0.94$. We can see that the model is able to fit the three stellar parameters well and fits T_{eff} and [Fe/H] better than log g. Simultaneously, we plot the scatter density plots of the predicted and true values on the test set (see the left column of Fig. 6). The green dashed line indicates the first-degree polynomial fit curve of the predicted and true values, and the red line is the image of *predicted value = true value*. The closer the green dashed line is to the red solid line, the better the prediction results. From the figure, we can state that the fitting results of T_{eff} and [Fe/H] are very close to the line of *predicted value = true value*, while the results of log g are relatively poor, which shows that the proposed CNN model has a better prediction for T_{eff} and [Fe/H], while log g is relatively more difficult to estimate. In addition, the right

column of Fig. 6 illustrates the variation of the residuals (true value-predicted value) with respect to the true values. The red line can show us more explicitly the turbulence of the residuals around zero.

Furthermore, experiments can be conducted on 10 008 VMP stars and 16 638 non-VMP stars separately to test whether the CNN model has a significant difference in measuring the parameters of VMP stars and those of non-VMP stars. The prediction results on the two test sets involving 3 003 VMP stars and 4 992 non-VMP stars are listed in Table 2. Briefly, the MAE values for log g, and 0.17 dex for [Fe/H] for the VMP stars, and 75.84 K for T_{eff} , 0.11 dex for log g, and 0.08 dex for [Fe/H] for the non-VMP stars. We can clearly demonstrate that the proposed CNN model is much better at deriving the parameters of non-VMP stars than VMP stars, which specifies the necessity to develop a model that can effectively measure the parameters of VMP stars.

With the results obtained above, we can conclude that the proposed CNN model has good accuracy in estimating stellar atmospheric parameters, which suggests that we can use the method for VMP star identification. By analysing the metallicity of the total dataset containing VMP and non-VMP stars, setting the label of VMP stars with [Fe/H]<-2.0 to 1 and the label of non-VMP stars with [Fe/H]>-2.0 to 0, we find 2 999 VMP stars and 4 995 non-VMP stars in the test set. The confusion matrix of the true and predicted values is shown in Fig. 7. Then we can calculate the precision, recall, and accuracy of the proposed CNN model for predicting VMP stars (see Table 3). Among the 2 966 stars predicted to be VMP stars, 2 811 stars are true VMP stars, with a precision rate of 94.77% (2 811/2 966); among the test set including 2 999 VMP stars, 2 811 stars are correctly predicted to be VMP stars, with a recall rate of 93.73% (2 811/2 999). Overall, our CNN model is also able to classify VMP stars and non-VMP stars well, with an accuracy of 95.70%.

4.2. Estimating [C/Fe] and identifying CEMP stars using the VMP stars dataset

We perform experiments using the dataset obtained by cross-matching in Section 2.1, which includes 8 117 VMP stars with [C/Fe] values. The best prediction results obtained on the test set are MAE=0.26 dex, STD=0.38 dex, and $R^2=0.64$. We also plot the scatter density plots of the predicted and true values of [C/Fe] and the residuals (Fig. 8). It can be seen from the figures that the proposed CNN model is also able to predict [C/Fe] well. By analysing the prediction results, we find that 240 stars are correctly predicted as CEMP stars, with a precision rate of 75.7% (240/317). The total accuracy rate is around 87.56%, indicating our model has good performance in classifying CEMP stars.

4.3. Estimating stellar parameters using the MARCS synthetic spectra including 9 644 stars

This section shows the outcomes of predicting the stellar parameters of the 9 644 MARCS spectra using the CNN model. The results on the test set of 2 894 stars are shown in Table 4. For T_{eff} , MAE=53.03 K, STD=80.78 K, $R^2=0.998$; for log g, MAE=0.056 dex, STD=0.097 dex, $R^2=0.995$; and for [Fe/H], MAE=0.047 dex, STD=0.093 dex, $R^2=0.995$. Compared to the

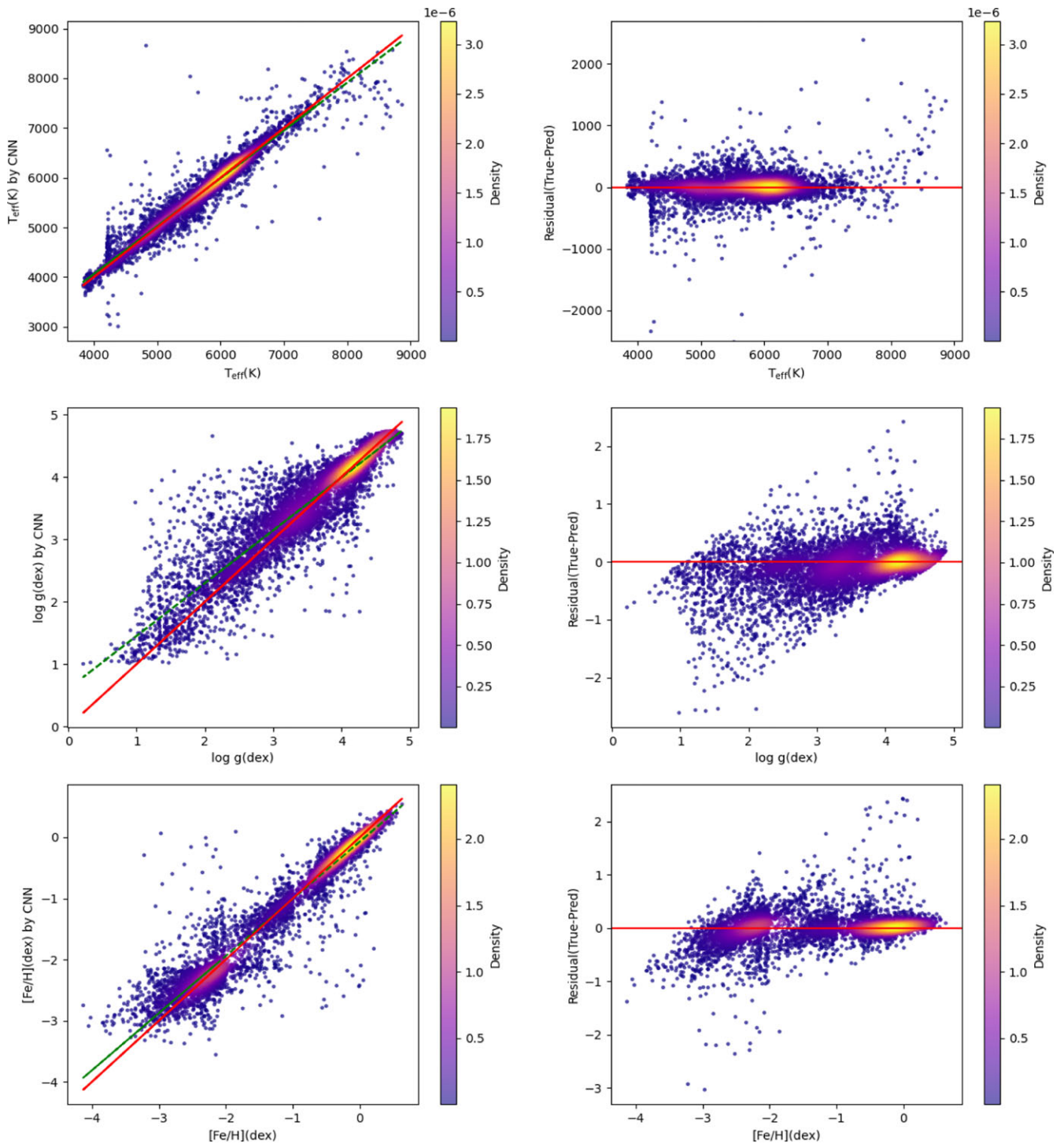


Figure 6. The left panel shows the true and predicted values of T_{eff} (top side), $\log g$ (centre), and $[\text{Fe}/\text{H}]$ (bottom side) obtained on the LAMOST test set. The red solid line is the plot of the function $y = x$. The green dashed line represents the first-degree polynomial fit curve of the predicted values to the true values. The right panel are their residuals against the true values.

errors obtained using the LAMOST dataset, the results for the synthetic spectra are much smaller, and in particular the R^2 values very close to 1 indicate a good fit of our model. The scatter density plots between the true and predicted values can be seen in Fig. 9, revealing there is only a little deviation between them.

5. Discussion

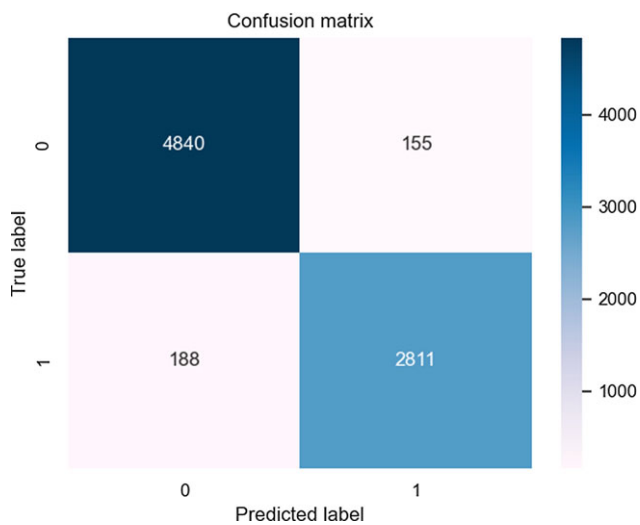
To further verify the effectiveness of the proposed CNN model, we introduce Random Forest (RF) and Support Vector Machine (SVM) algorithms to make comparisons. The dataset used in the comparison experiment is the same as that used in Section 4.1,

Table 2. The prediction results of the three fundamental atmospheric parameters on VMP star test set and non-VMP stars test set.

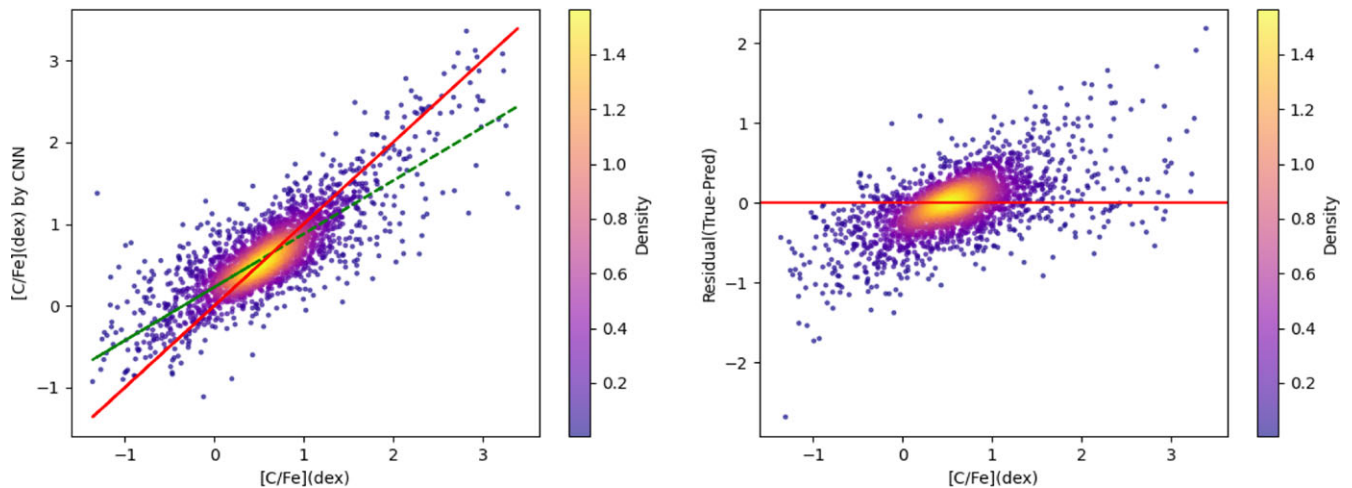
	$T_{\text{eff}}(\text{K})$			$\log g(\text{dex})$			$[\text{Fe}/\text{H}](\text{dex})$		
	MAE	STD	R^2	MAE	STD	R^2	MAE	STD	R^2
VMP	118.26	173.61	0.91	0.31	0.43	0.67	0.17	0.24	0.84
COMMON	75.84	156.14	0.96	0.11	0.18	0.95	0.08	0.13	0.94

Table 3. The results of classifying the VMP stars on test set including 7 994 stars.

Precision	Recall	Accuracy
94.77%	93.73%	95.70%

**Figure 7.** The confusion matrix of classifying the VMP stars on test set including 7 994 stars.

which includes a total of 26 646 spectral data and fundamental stellar parameters of the VMP stars and non-VMP stars. The training and test sets are also selected in line with the previous experiments to test whether the CNN model outperforms other algorithms.

**Figure 8.** The left panel shows the true and predicted values of $[\text{C}/\text{Fe}]$ obtained on the test set including 2 436 stars. The red solid line is the plot of the function $y = x$. The green dashed line represents the first-degree polynomial fit curve of the predicted values to the true values. The right panel is the residual against the true values.**Table 4.** The prediction results of the three fundamental atmospheric parameters on MARCS test set including 2 894 stars.

Parameter	MAE	STD	R^2
$T_{\text{eff}}(\text{K})$	53.03	80.78	0.998
$\log g(\text{dex})$	0.056	0.097	0.995
$[\text{Fe}/\text{H}](\text{dex})$	0.047	0.093	0.995

1. RF:

The RF algorithm (Breiman 2001) is a specific implementation of the bagging method, where multiple decision trees are trained and all results are combined. For regression problems, the prediction of the Random Forest is the average of all decision tree results. The advantage that this method can operate efficiently on large data sets and is not prone to overfitting has made it widely used in astronomical data analysis (Wang et al. 2019; Mahmudunnobe et al. 2021).

The *RandomForestRegressor* function in the *Scikit-learn* package in Python is imported to carry out experiments. To construct the optimal RF model, we tune the parameters for the number of decision trees (*n_estimators*) and the maximum number of features (*max_features*). The *GridSearchCV* function in Python provides us with a convenient method to automatically derive the optimal parameters and the score. The specific parameter-tuning process is detailed in the Appendix. The MAE values on the test set are, 122.57 K for T_{eff} , 0.30 dex for $\log g$, and 0.26 dex for $[\text{Fe}/\text{H}]$. The precision, recall, and accuracy of classifying the VMP stars are 93.12%, 74.02%, and 88.20%, respectively.

2. SVM:

SVM (Cortes and Vapnik 1995) is a binary classification model, which is essentially an optimization algorithm for solving convex quadratic programming problems. In addition to classification problems, SVM can also be applied to regression problems (SVR), which centres on finding a regression plane such that all data in a set are closest to

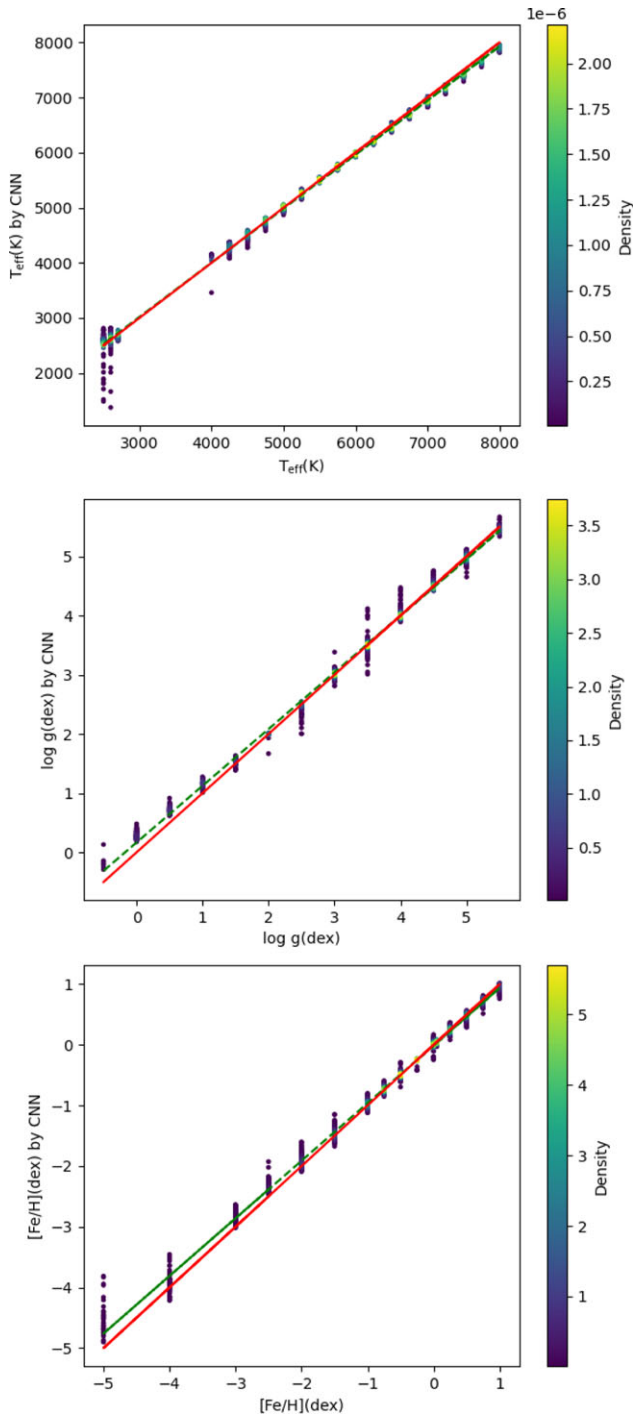


Figure 9. The true and predicted values of T_{eff} (top panel), $\log g$ (middle panel), and $[\text{Fe}/\text{H}]$ (bottom panel) on MARCS test set including 2 894 stars.

this plane. For non-linear regression problems, SVM can introduce a kernel function that turns the problem into an approximate linear regression problem.

To construct the optimal regression model, we use the SVR function of the *Scikit-learn* library in Python, utilizing the third-degree polynomial kernel function for training and tuning two relatively important parameters C and γ . C is the penalty factor of the target function and γ is the coefficient of the kernel function. The tuning

Table 5. The prediction results of the three fundamental atmospheric parameters on the test set including 7,994 stars using RF, SVM, and CNN methods.

	T_{eff} (K)			$\log g$ (dex)			$[\text{Fe}/\text{H}]$ (dex)		
	MAE	STD	R^2	MAE	STD	R^2	MAE	STD	R^2
RF	122.57	200.15	0.92	0.30	0.45	0.75	0.26	0.41	0.85
SVM	122.07	211.92	0.91	0.26	0.40	0.80	0.23	0.37	0.87
CNN	99.40	183.33	0.93	0.22	0.35	0.84	0.14	0.26	0.94

process is detailed in the Appendix. By using the optimal SVM model, the MAE values obtained on the test set are, 122.07 K for T_{eff} , 0.26 dex for $\log g$, and 0.23 dex for $[\text{Fe}/\text{H}]$. The precision, recall, and accuracy of classifying the VMP stars are 95.37%, 77.66%, and 90.21%, respectively.

The specific results of the comparison experiments are shown in Table 5. The MAE, STD, and R^2 values for estimating stellar parameters using the three machine learning methods are included. We can see that CNN estimates stellar parameters with higher accuracy than the other two methods from all three metrics.

We also draw a bar chart of the results of identifying VMP stars with these three methods. From Fig. 10, we can clearly demonstrate that although the precision of the three methods is comparable, the recall rates of RF and SVM are much lower than that of the CNN, which indicates that the probability of VMP stars being predicted as common stars can be greatly reduced using the proposed CNN model. In terms of accuracy, the CNN model is also able to better classify VMP stars and common stars.

Additionally, Wang et al. (2022) carried out a fairly similar exercise to ours using neural networks. Similarly, they utilized the VMP stars catalog from Li et al. (2018) as a comparative reference. Employing their own method, they determined the $[\text{Fe}/\text{H}]$ values within the sample. Their investigation yielded Standard Deviation (σ) values of 0.299 dex and 0.219 dex for two kinds of $[\text{Fe}/\text{H}]$ values ($[\text{Fe}/\text{H}]$ -NN-PASTEL and $[\text{Fe}/\text{H}]$ -NN-VMP, refer to Figure 21 in their study). In contrast to our VMP star results with an STD value of 0.24 dex (as shown in Table 2), the error of their improved $[\text{Fe}/\text{H}]$ values ($[\text{Fe}/\text{H}]$ -NN-VMP) is slightly better than ours, but the spectral resolution used by us is extremely low at only 200, which is much lower than the LAMOST low-resolution spectra ($R \sim 1\ 800$) employed by them. Consequently, it can be inferred that our model capably estimates stellar parameters even for spectra possessing a resolution as low as 200.

6. Conclusion

This paper investigates the effectiveness of the CNN model in estimating stellar parameters for low-resolution spectra ($R \sim 200$) and the ability to identify VMP stars. We constructed a two-dimensional CNN model consisting of three convolutional and two fully connected layers and selected a catalog including 10 008 VMP stars and 16 638 non-VMP stars for our experiments. The resolution of these stellar spectra was scaled down from $R \sim 1\ 800$ to $R \sim 200$ to match the CSST's spectral data, and then the spectral data with 410 features could be derived through interpolation and normalization. By collapsing these one-dimensional spectra into two-dimensional matrices and feeding them into the CNN model, we successfully estimated corresponding stellar parameters. The results show that for T_{eff} , MAE=99.40 K, STD=183.33 K,

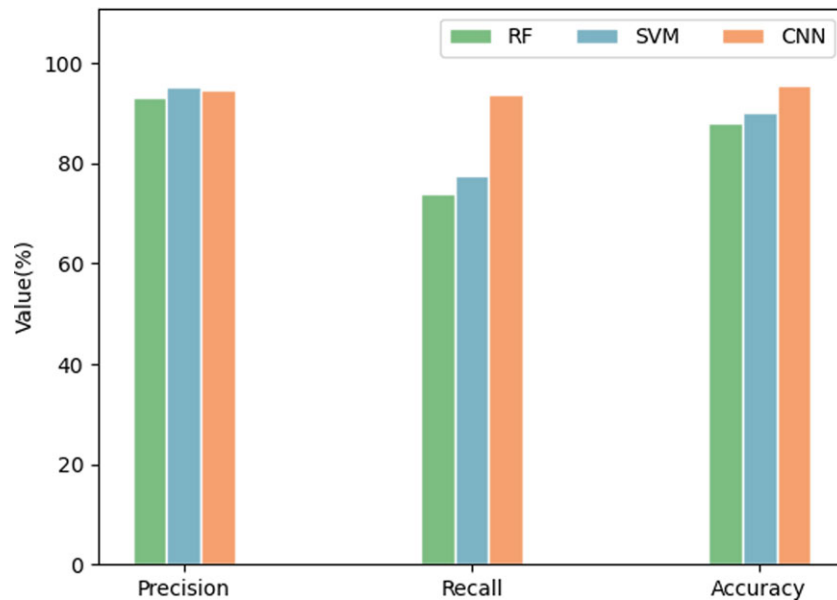


Figure 10. The bar chart of the precision rate, recall rate, and accuracy of identifying VMP stars using RF (green), SVM (blue), and CNN (orange).

$R^2=0.93$; for $\log g$, $MAE=0.22$ dex, $STD=0.35$ dex, $R^2=0.84$; for $[Fe/H]$, $MAE=0.14$ dex, $STD=0.26$ dex, $R^2=0.94$; and for $[C/Fe]$, $MAE=0.26$ dex, $STD=0.37$ dex, $R^2=0.64$. While the CNN model exhibited slightly diminished performance in deriving parameters of the VMP stars compared to non-VMP stars, it was still able to distinguish VMP stars with a precision rate of 94.77%, a recall rate of 93.73% and an accuracy of 95.70%. Impressively, the model also effectively identified CEMP stars in VMP stars, achieving an accuracy of 87.56%. Moreover, we illustrated the superiority of the CNN model over the RF and SVM algorithms in that it can predict stellar parameters with higher accuracy and identify VMP stars better, with a recall rate nearly 20% higher than the other two approaches. The efficiency of the CNN model was also tested on the MARCS synthetic spectra, and the MAE values obtained on the test set were 53.03 K for T_{eff} , 0.056 dex for $\log g$, and 0.047 dex for $[Fe/H]$.

To sum up, the CNN model proposed in this paper can productively measure the stellar parameters of spectra with a resolution of 200 and excels in identifying VMP stars. This work lays a robust foundation for future investigations of a large number of low-resolution spectra obtained by the CSST and searching for VMP stars from them. This will not only greatly expand the VMP star candidates, but also lead to a better understanding of the evolution of the Milky Way.

Acknowledgement. This work is supported by the National Natural Science Foundation of China under grant numbers 11873037, 11603012, and 11603014 and partially supported by the Young Scholars Program of Shandong University, Weihai (2016WHWLJH09), and the science research grants from the China manned Space Project with No CMS-CSST-2021-B05 and CMS-CSST-2021-A08.

References

Aoki, W., Beers, T. C., Christlieb, N., Norris, J. E., Ryan, S. G., & Tsangarides, S. 2007, *ApJ*, **655**, 492. <https://doi.org/10.1086/509817>. arXiv: [astro-ph/0609702](https://arxiv.org/abs/astro-ph/0609702) [astro-ph].

- Arentsen, A., et al. 2020, *MNRAS*, **496**, 4964. <https://doi.org/10.1093/mnras/staa1661>. arXiv: [2006.08641](https://arxiv.org/abs/2006.08641) [astro-ph.GA]
- Barklem, P. S., et al. 2005, *A&A*, **439**, 129
- Beers, T. 2011, *HST Proposal ID 12554*. Cycle 19
- Beers, T. C., & Christlieb, N. 2005, *ARA&A*, **43**, 531. <https://doi.org/10.1146/annurev.astro.42.053102.134057>
- Blanco-Cuaresma, S., Soubiran, C., Heiter, U., & Jofrè, P. 2014, *A&A*, **569**, A111. <https://doi.org/10.1051/0004-6361/201423945>. arXiv: [1407.2608](https://arxiv.org/abs/1407.2608) [astro-ph.IM]
- Breiman, L. 2001, *ML*, **45**, 5. <https://doi.org/10.1023/A:1010933404324>
- Christlieb, N., Schörck, T., Frebel, A., Beers, T. C., Wisotzki, L., & Reimers, D. 2008, *A&A*, **484**, 721. <https://doi.org/10.1051/0004-6361:20078748>. arXiv: [0804.1520](https://arxiv.org/abs/0804.1520) [astro-ph]
- Cortes, C., & Vapnik, V. 1995, *ML*, **20**, 273
- Da Costa, G. S., et al. 2019, *MNRAS*, **489**, 5900. <https://doi.org/10.1093/mnras/stz2550>. arXiv: [1909.06227](https://arxiv.org/abs/1909.06227) [astro-ph.SR]
- Du, B., et al. 2021, *RAA*, **21**, 202. <https://doi.org/10.1088/1674-4527/21/8/202>
- Fabbro, S., Venn, K. A., OŠbriain, T., Bialek, S., KIELTY, C. L., Jahandar, F., & Monty, S. 2018, *MNRAS*, **475**, 2978. <https://doi.org/10.1093/mnras/stx3298>. arXiv: [1709.09182](https://arxiv.org/abs/1709.09182) [astro-ph.IM]
- Frebel, A. 2018, *ARNPS*, **68**, 237. <https://doi.org/10.1146/annurev-nucl-101917-021141>. arXiv: [1806.08955](https://arxiv.org/abs/1806.08955) [astro-ph.SR]
- Frebel, A., et al. 2006, *ApJ*, **652**, 1585. <https://doi.org/10.1086/508506>. arXiv: [astro-ph/0608332](https://arxiv.org/abs/astro-ph/0608332) [astro-ph]
- Frebel, A., & Norris, J. E. 2015, *ARA&A*, **53**, 631. <https://doi.org/10.1146/annurev-astro-082214-122423>. arXiv: [1501.06921](https://arxiv.org/abs/1501.06921) [astro-ph.SR]
- García Pérez, A. E., et al. 2013, *ApJL*, **767**, L9. <https://doi.org/10.1088/2041-8205/767/1/L9>. arXiv: [1301.1367](https://arxiv.org/abs/1301.1367) [astro-ph.SR]
- Gavel, A., Andrae, R., Fouesneau, M., Korn, A. J., & Sordo, R. 2022, Estimating $left[\alpha/xtFeight]$ from gaia low-resolution bp/rp spectra using the extra-trees algorithm
- Gustafsson, B., Edvardsson, B., Eriksson, K., Jørgensen, U. G., Nordlund, Å., & Plez, B. 2008, *A&A*, **486**, 951. <https://doi.org/10.1051/0004-6361:200809724>. arXiv: [0805.0554](https://arxiv.org/abs/0805.0554) [astro-ph]
- Hinton, G. E. 2008, Reducing the dimensionality of data with neural
- Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. 2012, arXiv e-prints (July): [arXiv:1207.0580](https://arxiv.org/abs/1207.0580). <https://doi.org/10.48550/arXiv.1207.0580>. arXiv: [1207.0580](https://arxiv.org/abs/1207.0580) [cs.NE]
- Ioffe, S., & Szegedy, C. 2015, International Conference on Machine Learning, 448. pmlr
- Kingma, D. P., & Ba, J. 2014, arXiv preprint arXiv: [1412.6980](https://arxiv.org/abs/1412.6980)

- Koleva, M., Prugniel, Ph., Bouchard, A., & Wu, Y. 2009, *A&A*, **501**, 1269. <https://doi.org/10.1051/0004-6361/200811467>. arXiv: 0903.2979 [astro-ph.IM]
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. 2012, *Advances in Neural Information Processing Systems* 25
- Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. 1998, *Proc. IEEE*, **86**, 2278. <https://doi.org/10.1109/5.726791>
- Lee, Y. S., et al. 2013, *AJ*, **146**, 132. <https://doi.org/10.1088/0004-6256/146/5/132>. arXiv: 1310.3276 [astro-ph.SR]
- Lee, Y. S., et al. 2008, *AJ*, **136**, 2022. <https://doi.org/10.1088/0004-6256/136/5/2022>. arXiv: 0710.5645 [astro-ph]
- Leung, H. W., & Bovy, J. 2019, *MNRAS*, **483**, 3255. <https://doi.org/10.1093/mnras/sty3217>. arXiv: 1808.04428 [astro-ph.GA]
- Li, H., Tan, K., & Zhao, G. 2018, *ApJS*, **238**, 16. <https://doi.org/10.3847/1538-4365/aada4a>. arXiv: 1809.03881 [hep-ph]
- Li, H. N., Zhao, G., Christlieb, N., Wang, L., Wang, W., Zhang, Y., Hou, Y., & Yuan, H. 2015, *ApJ*, **798**, 110. <https://doi.org/10.1088/0004-637X/798/2/110>. arXiv: 1501.03062 [astro-ph.SR]
- Li, X., Zeng, S., Wang, Z., Du, B., Kong, X., & Liao, C. 2022, *MNRAS*, **514**, 4588. <https://doi.org/10.1093/mnras/stac1625>. arXiv: 2207.06042 [astro-ph.IM]
- Lu, Y., Li, X. R., Lin, Y. T., & Qiu, K. B. 2018, *AAS*, **59**, 35
- Lucatello, S., Beers, T. C., Christlieb, N., Barklem, P. S., Rossi, S., Marsteller, B., Sivarani, T., & Lee, Y. S. 2006, *ApJL*, **652**, L37. <https://doi.org/10.1086/509780>. arXiv: astro-ph/0609730 [astro-ph]
- Mahmudunnobe, Md., Hasan, P., Raja, M., & Hasan, S. N. 2021, *EPJST*, **230**, 2177. <https://doi.org/10.1140/epjs/s11734-021-00205-x>. arXiv: 2103.05826 [astro-ph.SR]
- Marsteller, B., Beers, T. C., Rossi, S., Christlieb, N., Bessell, M., & Rhee, J. 2005, *NPhA*, **758**, 312. <https://doi.org/abs/10.1016/j.nuclphysa.2005.05.056>. arXiv: astro-ph/0408380 [astro-ph]
- Matijević, G., et al. 2017, *A&A*, **603**, A19
- Ness, M., Hogg, D. W., Rix, H.-W., Ho, A. Y. Q., & Zasowski, G. 2015, *ApJ*, **808**, 16. <https://doi.org/10.1088/0004-637X/808/1/16>. arXiv: 1501.07604 [astro-ph.SR]
- Placco, V. M., Frebel, A., Beers, T. C., & Stancliffe, R. J. 2014, *ApJ*, **797**, 21. <https://doi.org/10.1088/0004-637X/797/1/21>. arXiv: 1410.2223 [astro-ph.SR]
- Rasmussen, K. C., Zepeda, J., Beers, T. C., Placco, V. M., Depagne, É., Frebel, A., Dietz, S., & Hartwig, T. 2020, *ApJ*, **905**, 20. <https://doi.org/10.3847/1538-4357/abc005>. arXiv: 2010.04214 [astro-ph.SR]
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. 2016, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779.
- Reggiani, H., et al. 2019, *A&A*, **627**, A177. <https://doi.org/10.1051/0004-6361/201935156>. arXiv: 1906.08281 [astro-ph.SR]
- Rhee, J., Beers, T. C., & Irwin, M. J. 2001, in *American Astronomical Society Meeting Abstracts*, 199:91.08. American Astronomical Society Meeting Abstracts. December
- Roederer, I. U., Preston, G. W., Thompson, I. B., Shectman, S. A., Sneden, C., Burley, G. S., & Kelson, D. D. 2014, *AJ*, **147**, 136. <https://doi.org/10.1088/0004-6256/147/6/136>. arXiv: 1403.6853 [astro-ph.SR]
- Ronneberger, O., Fischer, P., & Brox, T. 2015, in *Medical Image Computing and Computer-Assisted Intervention—Miccai 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III (Vol. 18; Springer)*, 234.
- Salsi, A., Nardetto, N., Plez, B., & Mourard, D. 2022, *A&A*, **662**, A120. <https://doi.org/10.1051/0004-6361/202142133>. arXiv: 2203.16320 [astro-ph.SR]
- Szegedy, C., et al. 2015, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1
- Ting, Y.-S., Conroy, C., Rix, H.-W., & Cargile, P. 2019, *ApJ*, **879**, 69. <https://doi.org/10.3847/1538-4357/ab2331>. arXiv: 1804.01530 [astro-ph.SR]
- VandenBerg, D. A., Edvardsson, B., Casagrande, L., & Ferguson, J. W. 2021, *MNRAS*, **509**, 4189. ISSN: 0035-8711. <https://doi.org/10.1093/mnras/stab2996>. eprint: <https://academic.oup.com/mnras/article-pdf/509/3/4189/41505523/stab2996.pdf>.
- Wang, C., Huang, Y., Yuan, H. B., Zhang, H. W., Xiang, M. S., & Liu, X. W. 2022, *ApJS*, **259**, 51. <https://doi.org/10.3847/1538-4365/ac4df7>. arXiv: 2201.09442 [astro-ph.SR]
- Wang, J., Zhen-Ping, Y. I., Li-Li, Y., Hui-Fen, D., Jing-Chang, P., Yu-De, B. U., & School Of Mechanical, Electrical Amp, Information Engineering, and Shandong University. 2019, *SSA*
- Wang, R., et al. 2020, *ApJ*, **891**, 23. <https://doi.org/10.3847/1538-4357/ab6dea>. arXiv: 2001.03470 [astro-ph.SR]
- Witten, C. E. C., et al. 2022, *MNRAS*, **516**, 3254. <https://doi.org/10.1093/mnras/stac2273>. arXiv: 2205.12271 [astro-ph.SR]
- Wu, Y., Du, B., Luo, A., Zhao, Y., & Yuan, H. 2014, in *Statistical Challenges in 21st Century Cosmology*, ed. A. Heavens, J.-L. Starck, & A. Krone-Martins (Vol. 306), 340. <https://doi.org/10.1017/S1743921314010825>. arXiv: 1407.1980 [astro-ph.IM]
- Wu, Y., Luo, A. L., Shi, J. R., Bai, Z. R., & Zhao, Y. H. 2010, *New metal-poor star candidates from guo shou-jing telescope (lamost) commissioning observations*
- Yuan, H.-B., Deng, D.-S., & Sun, Y. 2021, *RAA*, **21**, 074. <https://doi.org/10.1088/1674-4527/21/3/074>. arXiv: 2010.14005 [astro-ph.IM]
- Yuan, Z., et al. 2020, *ApJ*, **891**, 39. <https://doi.org/10.3847/1538-4357/ab6ef7>. arXiv: 1910.07538 [astro-ph.GA]
- Zhan, H. 2021, *CSB*, **66**, 1290

Appendix A. Parameter-tuning process of RF and SVM

1. RFr

The RF model can be regarded as a decision tree model embedded into the bagging framework, so we first perform parameter selection on the outer bagging framework ($n_{estimators}$) and then on the inner decision tree model ($max_features$). When optimizing a certain parameter, the other parameters need to be set as constants. The parameter-tuning process is performed on T_{eff} , $\log g$, and $[Fe/H]$, respectively. For T_{eff} , when default values are used for all parameters, the best score is 0.914. After that, we perform a ten-fold cross-validation. The range of $n_{estimators}$ is chosen to be 1–150 and the step size is 10. The best $n_{estimators}$ obtained is 110 and the score is 0.916. On the basis of $n_{estimators}$ of 110, the $max_features$ range is chosen to be 1–100 and the step size is 1. The best $max_features$ obtained is 95, with a score of 0.918. It can be seen that there is a small increase in the score, indicating the parameter-tuning process is effective. Following the same steps, we obtain the optimal parameters for $\log g$ is $n_{estimators}=120$, $max_features=94$ with a score of 0.743, and for $[Fe/H]$ is $n_{estimators}=110$, $max_features=52$ with the score of 0.851.

2. SVM

The tuning process for the SVM model is performed separately for T_{eff} , $\log g$, and $[Fe/H]$. We use a grid search for 5-fold cross-validation, first tuning C , and then fixing the optimal C value to adjust $gamma$. C is set to 0.1, 1, 10, and $gamma$ is set to 0.001, 0.01, 0.1, and ‘scale’ ($gamma = 1/(n_features * X.var())$, where $n_features$ is the number of the input features, $X.var()$ is the variance of the input features). The experimental results show that for T_{eff} , $\log g$, and $[Fe/H]$, the optimal C values are all 10, with scores of 0.87, 0.79, and 0.87, respectively. On this basis, optimal $gamma$ values are obtained as 0.1, ‘scale’, and ‘scale’, with scores of 0.91, 0.81, and 0.88, respectively. Using the obtained optimal SVM model, we can conduct subsequent experiments.