RAA 2017 Vol. **17** No. **4**, 36 (8pp) doi: 10.1088/1674–4527/17/4/36

http://www.raa-journal.org *http://iopscience.iop.org/raa*

***R**esearch in*
***A**stronomy and*
***A**strophysics*

Parameterizing Stellar Spectra Using Deep Neural Networks

Xiang-Ru Li[1], Ru-Yang Pan[1] and Fu-Qing Duan[2]

[1] School of Mathematical Sciences, South China Normal University, Guangzhou 510631, China; *xiangru.li@gmail.com*

[2] College of Information Science and Technology, Beijing Normal University, Beijing 100875, China

Received 2016 November 17; accepted 2017 January 3

Abstract Large-scale sky surveys are observing massive amounts of stellar spectra. The large number of stellar spectra makes it necessary to automatically parameterize spectral data, which in turn helps in statistically exploring properties related to the atmospheric parameters. This work focuses on designing an automatic scheme to estimate effective temperature ($T_{\rm eff}$), surface gravity ($\log g$) and metallicity [Fe/H] from stellar spectra. A scheme based on three deep neural networks (DNNs) is proposed. This scheme consists of the following three procedures: first, the configuration of a DNN is initialized using a series of autoencoder neural networks; second, the DNN is fine-tuned using a gradient descent scheme; third, three atmospheric parameters $T_{\rm eff}$, $\log g$ and [Fe/H] are estimated using the computed DNNs. The constructed DNN is a neural network with six layers (one input layer, one output layer and four hidden layers), for which the number of nodes in the six layers are 3821, 1000, 500, 100, 30 and 1, respectively. This proposed scheme was tested on both real spectra and theoretical spectra from Kurucz's new opacity distribution function models. Test errors are measured with mean absolute errors (MAEs). The errors on real spectra from the Sloan Digital Sky Survey (SDSS) are 0.1477, 0.0048 and 0.1129 dex for $\log g$, $\log T_{\rm eff}$ and [Fe/H] (64.85 K for $T_{\rm eff}$), respectively. Regarding theoretical spectra from Kurucz's new opacity distribution function models, the MAE of the test errors are 0.0182, 0.0011 and 0.0112 dex for $\log g$, $\log\ T_{\rm eff}$ and [Fe/H] (14.90 K for $T_{\rm eff}$), respectively.

Key words: methods: statistical — methods: data analysis — stars: fundamental parameters — stars: atmospheres — stars: abundances — techniques: spectroscopic

1 INTRODUCTION

Some large-scale sky surveys are observing and will collect massive amounts of stellar spectra, for example, the Sloan Digital Sky Survey (SDSS; York et al. 2000; Alam et al. 2015; Ahn et al. 2012), Large Sky Area Multi-Object Fiber Spectroscopic Telescope/Guo Shou Jing Telescope (LAMOST; Zhao et al. 2006; Luo et al. 2015; Cui et al. 2012), and *Gaia*-ESO Survey (Gilmore et al. 2012; Randich et al. 2013). The large number of stellar spectra makes it necessary to automatically parameterize the spectra, which will in turn help statistical investigations of problems related to atmospheric parameters.

The present work studies the problem of spectrum parameterization. A typical class of schemes is based on (feedforward) neural networks ((F)NNs: Willemsen et al. 2005; Giridhar et al. 2006; Re Fiorentin et al. 2007; Gray et al. 2009; Tan et al. 2013a). In these NNs, the information moves in only one direction, that is from the input nodes (neurons), through the hidden nodes, and to the output nodes (neurons). In atmospheric parameter estimation, the input nodes represent a stellar spectrum, and the output node(s) represent(s) the atmospheric parameter(s) to be estimated, e.g., effective temperature $T_{\rm eff}$, surface gravity $\log\ g$ and metallicity [Fe/H]. An NN is commonly obtained by a back-propagation (BP) algorithm (Rumelhart et al. 1986).

For example, Bailer-Jones (2000) studied the prediction accuracy of effective temperature $T_{\rm eff}$, surface gravity $\log\ g$ and metallicity [Fe/H] using an FNN with two hidden layers on theoretical spectra with various resolutions and signal-to-noise ratios. Snider et al. (2001) parameterized medium-resolution spectra of F- and G-type stars using two FNN networks with one and two hidden

layers respectively. Manteiga et al. (2010) investigated the estimation of atmospheric parameters from stellar spectra by extracting features using time-frequency decomposition techniques and an FNN with one hidden layer. Li et al. (2014) investigated the atmospheric parameter estimation problem by detecting spectral features by LASSO first and subsequently estimating the atmospheric parameters using an FNN with one hidden layer.

This article investigates the spectrum parameterization problem using a deep NN (DNN). In application, a traditional NN usually has one or two hidden layers. By contrast, DNNs have two typical characteristics: (1) a DNN usually has more hidden layers; (2) two procedures are needed in estimating a DNN: prelearning and fine-tuning. This scheme has been studied extensively in artificial intelligence and data mining, and shows excellent performance in many applications, e.g., object recognition (Krizhevsky et al. 2012), speech recognition (Dahl et al. 2010; Hinton et al. 2012), pedestrian detection (Sermanet et al. 2013), image segmentation (Couprie et al. 2013), traffic sign classification (Ciresan et al. 2012), image transcription (Goodfellow et al. 2013), sequence to sequence learning (Sutskever et al. 2014) and machine translation (Bahdanau et al. 2014). This work investigated the application of this scheme in spectrum parameterization.

In this work, Section 2 introduces the NN, DNN, their learning algorithms and the proposed stellar parameter estimation scheme. Section 3 reports some experimental evaluations on real and synthetic spectra. Finally, our work is summarized in Section 4.

2 PARAMETERIZING STELLAR SPECTRA USING A DNN

2.1 A Neural Network (NN)

This work investigated a scheme to parameterize a stellar spectrum using a DNN. An NN consists of a series of neurons in multiple layers.

Figure 1 is a diagram of an NN with L layers. In this diagram, a solid circle represents a neuron, and a dashed circle is a bias unit used in describing the relationships between neurons.

In an NN, every neuron is a simple computational unit and has an input and an output, z and a, respectively. For example, $z_k^{(l)}$ and $a_k^{(l)}$ denote the input and output respectively of the k-th neuron in the l-th layer, where $l = 1, 2, \cdots, L$; $k = 1, \cdots, n_l$; and n_l represents the number of neurons in the l-th layer. The relationship between an input and an output is usually described by an activation function $g(\cdot)$ on layers $l = 2, \cdots, L-1$

$$a = g(z). \tag{1}$$

This work used the sigmoid function

$$g(z) = 1/(1+e^{-z}). \tag{2}$$

A neuron receives signals from every neuron in the previous layer as follows

$$z_k^{(l+1)} = \sum_{i=1}^{n_l} w_{ki}^{(l)} a_i^{(l)} + b_k^{(l)}, \tag{3}$$

where $l = 1, \cdots, L-1$, and $w_{ki}^{(l)}$ describe the relationship between the k-th and the i-th neurons on the $(l+1)$-th and l-th layers (this relationship is represented with a line between the two neurons in Figure 1), respectively; $b_k^{(l)}$ is the bias associated with the k-th neuron in the $(l+1)$-th layer (represented with a line between the k-th neuron and bias unit in the $(l+1)$-th and l-th layers, respectively), and n_l is the number of neurons in the l-th layer.

Generally, the first layer and the last layer are called input and output layers, respectively; the other layers are referred to as hidden layers. In the first layer and last layer, the output of a neuron is the same as its input

$$a_k^{(l)} = z_k^{(l)}, \quad k = 1, \cdots, n_1; \quad l = 1 \text{ and } L. \tag{4}$$

Suppose $\boldsymbol{x} = (x_1, \cdots, x_{n_1})^{\mathrm{T}}$ is a representation of a signal (e.g., a stellar spectrum). If $\boldsymbol{x}$ is an input into an NN in Figure 1 by letting $\boldsymbol{z}^{(1)} = \boldsymbol{x}$, then an output $\boldsymbol{a}^{(L)} = (a_1^{(L)}, \cdots, a_{n_L}^{(L)})$ can be computed from the last layer of this network (Eqs. (3) and (1)), where $\boldsymbol{z}^{(1)} = (z_1^{(1)}, \cdots, z_{n_1}^{(1)})^{\mathrm{T}}$. Therefore, an NN implements a non-linear mapping $h_{\boldsymbol{W},\boldsymbol{b}}(\cdot)$ from an input $\boldsymbol{x} = (x_1, \cdots, x_{n_1})^{\mathrm{T}}$ to an output $\boldsymbol{a}^{(L)}$ of the last layer

$$\boldsymbol{a}^{(L)} = h_{\boldsymbol{W},\boldsymbol{b}}(\boldsymbol{x}), \tag{5}$$

where

$$\boldsymbol{b} = \{\boldsymbol{b}^{(l)}\} \tag{6}$$

is the set of biases,

$$\boldsymbol{W} = \{\boldsymbol{W}^{(l)}, l = 1, \cdots, L\} \tag{7}$$

the set of the weights associated with an NN in Equation (3), $\boldsymbol{b}^l = \{b_j^{(l)}, 1 \le j \le n_l\}$ and $\boldsymbol{W}^{(l)} = \{W_{ji}^{(l)}\}$.

To define an NN, besides L, $\boldsymbol{W}$ and $\boldsymbol{b}$, one more set of parameters exists

$$(n_1, n_2, \cdots, n_L). \tag{8}$$

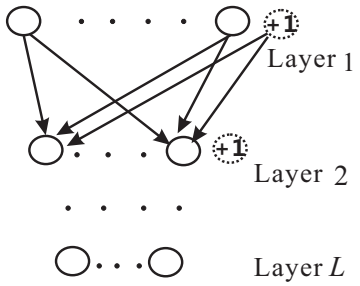


Fig. 1 A diagram of a neural network.

2.2 A BP Algorithm for Obtaining an NN

Let

$$S = \{(\boldsymbol{x}, \boldsymbol{y})\} \tag{9}$$

be a training set for an NN, where $\boldsymbol{x} = (x_1, \cdots, x_{n_1})^{\mathrm{T}}$ can be a representation of a spectrum and $\boldsymbol{y}$ is the expected output corresponding to $\boldsymbol{x}$. Section 3.1 discusses more about the training set.

In an NN, some parameters $\boldsymbol{W}$ and $\boldsymbol{b}$ should be given. These parameters are computed by minimizing an objective function $J(\boldsymbol{W}, \boldsymbol{b})$

$$\begin{aligned} J(\boldsymbol{W}, \boldsymbol{b}) = \ & \frac{1}{N} \sum_{\boldsymbol{x} \in S} (\frac{1}{2} \| h_{\boldsymbol{W}, \boldsymbol{b}}(\boldsymbol{x}) - \boldsymbol{y} \|^2) \\ & + \frac{\lambda}{2} \sum_{l=1}^{L-1} \sum_{i=1}^{n_l} \sum_{j=1}^{n_{l+1}} (w_{ji}^{(l)})^2, \end{aligned} \tag{10}$$

where N is the sample number of the training set and λ is a preset parameter with non-negative value controlling weight decay effects.

The first term of Equation (10) presents empirical evidence of inconsistences between the actual and expected outputs of an autoencoder; this term ensures the NN can be reconstructed. The second term is for regularization, which is used to reduce possible risks of overfitting to the training set by controlling model complexity.

To obtain our NN from a training set, we initialize each parameter $w_{ij}^{(l)}$ and $b_i^{(l)}$ to a small random value near zero; subsequently, two parameters $\boldsymbol{W}$ and $\boldsymbol{b}$ are iteratively optimized using a gradient descent method based on the objective function J in Equation (10). This learning scheme is referred to as a BP algorithm (Rumelhart et al. 1986; Ng et al. 2012).

2.3 Self-Taught Learning Applied to DNNs

In a BP algorithm, the parameters $\boldsymbol{W}$ and $\boldsymbol{b}$ are initialized with a small random value. However, the obtained results of the BP algorithm are unsatisfactory when the number of layers in an NN is higher than 4. In this case, $\boldsymbol{b} = \{\boldsymbol{b}^{(l)}\}$ and $\boldsymbol{W} = \{\boldsymbol{W}^{(l)}, l = 1, \cdots, L\}$ can be initialized using autoencoder networks.

An autoencoder is a specific kind of NN with three characteristics:

- There is a unique hidden layer. The number of neurons in this hidden layer is denoted by n_2^{ae}.
- The output layer has the same number of neurons as the input layer. This number of neurons in the input layer is denoted by n_1^{ae}.
- The expected outputs of the NN are also its inputs.

Therefore, the parameters of an autoencoder are $\boldsymbol{b}^{\mathrm{ae}}$, $\boldsymbol{W}^{\mathrm{ae}}$ and n^{ae}, where $\boldsymbol{b}^{\mathrm{ae}} = \{\boldsymbol{b}^{(1,\mathrm{ae})}, \boldsymbol{b}^{(2,\mathrm{ae})}\}$ is a set of biases, $\boldsymbol{W}^{\mathrm{ae}} = \{\boldsymbol{W}^{(1,\mathrm{ae})}, \boldsymbol{W}^{(2,\mathrm{ae})}\}$ a set of weights between neurons on different layers, and $n^{\mathrm{ae}} = (n_1^{\mathrm{ae}}, n_2^{\mathrm{ae}})$ the numbers of neurons on input layer and hidden layer.[1]

Therefore, to obtain a DNN (Fig. 1), the proposed learning scheme consists of the following processes:

- **Initialization using autoencoders.** To initialize the parameters $\boldsymbol{W}^{(1)}$ and $\boldsymbol{b}^{(1)}$ in Equations (7) and (6), an autoencoder with $(n_1^{\mathrm{ae}}, n_2^{\mathrm{ae}}) = (n_1, n_2)$ is established; $\boldsymbol{W}^{\mathrm{ae}} = \{\boldsymbol{W}^{(1,\mathrm{ae})}, \boldsymbol{W}^{(2,\mathrm{ae})}\}$ and $\boldsymbol{b}^{\mathrm{ae}} = \{\boldsymbol{b}^{(1,\mathrm{ae})}, \boldsymbol{b}^{(2,\mathrm{ae})}\}$ are obtained from a training set $S^{(1)} = \{(x, x), x \in S\}$ using the BP algorithm (Sect. 2.2) and let $\boldsymbol{W}^{(1)} = \boldsymbol{W}^{(1,\mathrm{ae})}$ and $\boldsymbol{b}^{(1)} = \boldsymbol{b}^{(1,\mathrm{ae})}$, where n_1 and n_2 are defined in Equation (8). To initialize $\boldsymbol{W}^{(l)}$ and $\boldsymbol{b}^{(l)}$, the training set S is input into the DNN in Figure 1 to produce the outputs $S^{(l)}$ from the l-th layer of the DNN in Figure 1. Subsequently, an autoencoder with $(n_1^{\mathrm{ae}}, n_2^{\mathrm{ae}}) = (n_l, n_{l+1})$ is established, $\boldsymbol{W}^{\mathrm{ae}} = \{\boldsymbol{W}^{(1,\mathrm{ae})}, \boldsymbol{W}^{(2,\mathrm{ae})}\}$ and $\boldsymbol{b}^{\mathrm{ae}} = \{\boldsymbol{b}^{(1,\mathrm{ae})}, \boldsymbol{b}^{(2,\mathrm{ae})}\}$ are obtained from the training set $S^{(l)}$ using the BP algorithm (Sect. 2.2), and the computed $\boldsymbol{W}^{(1,\mathrm{ae})}$ and $\boldsymbol{b}^{(1,\mathrm{ae})}$ are the initializations of $\boldsymbol{W}^{(1)}$ and $\boldsymbol{b}^{(1)}$, respectively, where $l = 2, \cdots, L$.
- **Fine-tuning.** The initialized $\boldsymbol{W}$ and $\boldsymbol{b}$ from the autoencoders are optimized using a gradient descent method based on the objective function J in

[1] The superscript 'ae' is an abbreviation of 'autoencoder'.

Equation (10) (this optimization procedure is the same as that in the BP algorithm: Section 2.2, Ng et al. 2012).

2.4 Spectrum Parameterization and Performance Evaluation

This work parameterizes stellar spectra using a DNN with six layers; its configurations of the DNN are $L = 6$ and $(n_1, \cdots, n_6) = (3821, 1000, 500, 100, 30, 1)$[2], where n_l is the number of neurons in the l-th layer of the DNN. In this DNN, the number of nodes in the input is equal to that of pixels of the spectrum to be processed. The three atmospheric parameters are estimated one by one, therefore the output layer has one node.

Before inputting into the DNN, a spectrum is normalized in this work. Suppose $\boldsymbol{x}$ is a spectrum. It is normalized as follows

$$x = \frac{x}{\sqrt{x^{\mathrm{T}} x}}, \tag{11}$$

where the superscript T is a transpose operation.

In the training set S in Equation (9), let $\boldsymbol{y}$ represent the effective temperature corresponding to a spectrum $\boldsymbol{x}$. From this training set S, a DNN estimator, namely $h_{W,h}$, can be obtained for estimating T_{eff}. Suppose that S' is a test set. In the present work, whether S' can be S or not is defined to introduce performance evaluation schemes.

Regarding S', the performance of the estimator $h_{W,h}$ is evaluated using the following three methods: mean absolute error (MAE), mean error (ME) and standard deviation (SD). They are defined as follows:

$$\mathrm{ME} = \frac{1}{M} \sum_{(\boldsymbol{x},\boldsymbol{y})\in S'} \mathrm{e}(\boldsymbol{x}, \boldsymbol{y}), \tag{12}$$

$$\mathrm{MAE} = \frac{1}{M} \sum_{(\boldsymbol{x},\boldsymbol{y})\in S'} |\mathrm{e}(\boldsymbol{x}, \boldsymbol{y})|, \tag{13}$$

$$\mathrm{SD} = \sqrt{\frac{1}{M} \sum_{(\boldsymbol{x},\boldsymbol{y})\in S'} (\mathrm{e}(\boldsymbol{x}, \boldsymbol{y}) - \mathrm{ME})^2}, \tag{14}$$

where M is the number of stellar spectra in S', and e is the deviation of the estimation from its reference value of the stellar parameter

$$\mathrm{e}(\boldsymbol{x}, \boldsymbol{y}) = h_{W,h}(\boldsymbol{x}) - \boldsymbol{y}\,. \tag{15}$$

These evaluation schemes are widely used in related researches (Re Fiorentin et al. 2007; Jofré et al. 2010; Tan et al. 2013b), and more about them is discussed in Li et al. (2015).

Similarly, the estimators for surface gravity $\log g$ and metallicity [Fe/H] are obtained and evaluated.

[2] This configuration is chosen based on experimental experiences using the training set.

3 EXPERIMENTS

This section evaluates the performance of the proposed scheme on both real stellar spectra and theoretical spectra.

3.1 Performance on SDSS Spectra

The experimental data set consists of 50 000 stellar spectra randomly selected from SDSS/SEGUE DR7 (Abazajian et al. 2009; Yanny et al. 2009). The signal-to-noise ratios of these spectra are [4.78397, 103.97] in the G band, [8.92085, 116.329] in the R band and [4.98563, 107.061] in the I band. The parameter ranges of these stellar spectra are presented in Table 1(a) and Figure 2, and their parameter reference values are obtained from the SDSS/SEGUE Spectroscopic Parameter Pipeline (SSPP; Beers et al. 2006; Lee et al. 2008a,b; Allende Prieto et al. 2008; Smolinski et al. 2011; Lee et al. 2011).

To parameterize the stellar spectra using the proposed DNN method, these spectra should be aligned based on rest wavelength. Therefore, all of these spectra are shifted to their rest frames and rebinned to a common wavelength range [3818.23, 9203.67] Å, and resampled in log(wavelength) with step size 0.0001.

The proposed scheme is a statistical method, DNN. The configuration, $\boldsymbol{W}$ and $\boldsymbol{b}$, of the proposed scheme should be estimated from some empirical data, and evaluated based on independent sets of observed stellar spectra. The two spectral sets are referred to as a training set and a test set, respectively. Therefore, we randomly select 20 000 spectra from the 50 000 stellar spectra as training samples, and the others as test samples.

Regarding the SDSS test spectra, the MAEs (mean absolute error defined in Eq. (13)) of the proposed DNN method are 64.85 K for effective temperature T_{eff} (0.0048 dex for $\log T_{\mathrm{eff}}$), 0.1129 dex for abundances [Fe/H] and 0.1477 dex for surface gravity $\log g$. To be comparable, therefore, the DNN is also evaluated using ME (mean error, Eq. (12)) and SD (standard deviation, Eq. (14)) (Table 2 (a)).

Some results are summarized in Table 2(b) from some related works in the literature. It is shown that the proposed DNN is accurate and excellent for stellar spectral parametrization.

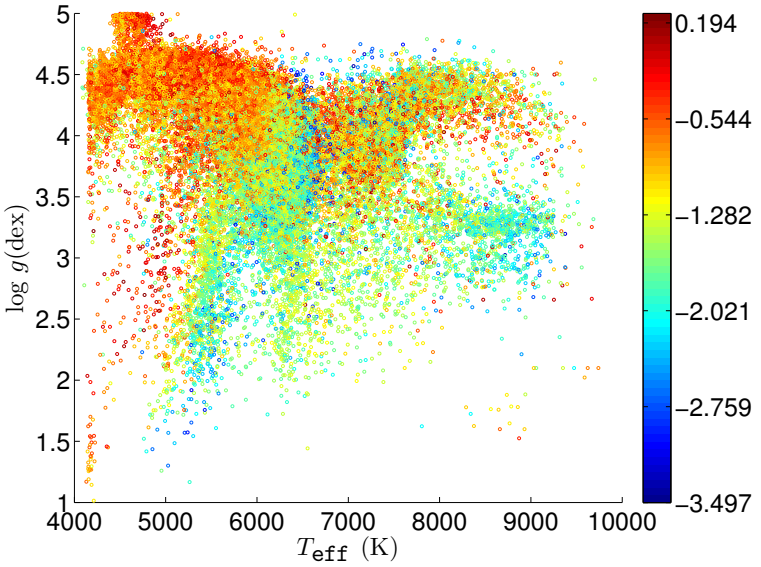


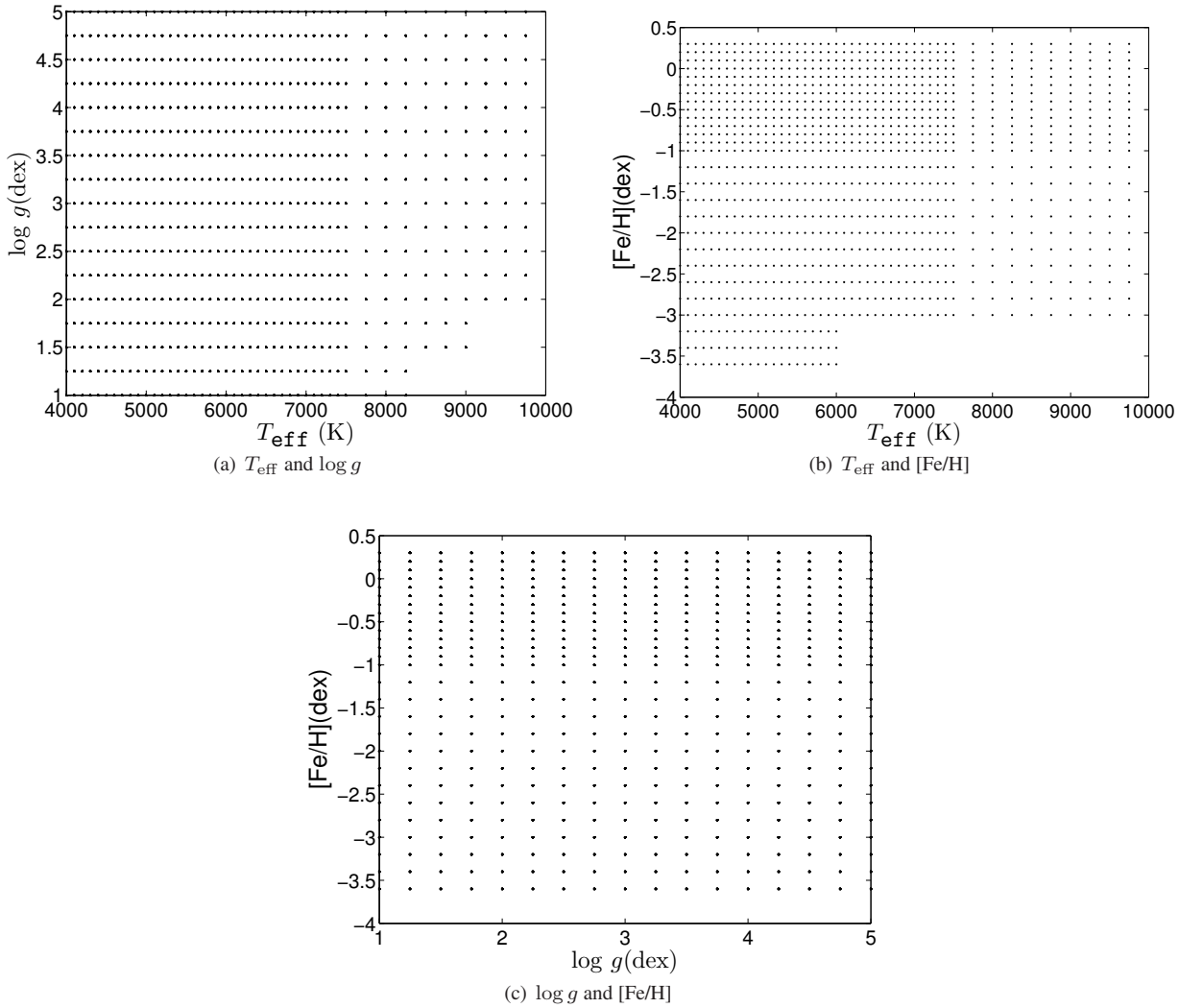
Fig. 2 Coverage of atmospheric parameters associated with the selected SDSS spectra. The color of the circles indicates the corresponding [Fe/H].

Table 1 Parameter Ranges of the Real Spectra

(a) Real spectra from SDSS DR7		(b) Theoretical spectra	
Atmospheric Parameters	Ranges	Atmospheric Parameters	Ranges
Effective Temperature $T_{\rm eff}$	[4088, 9740] K	Effective Temperature $T_{\rm eff}$	[4000, 9750] K
Surface Gravity $\log g$	[1.015, 4.998] dex	Surface Gravity $\log g$	[1, 5] dex
Metallicity [Fe/H]	[–3.497, 0.268] dex	Metallicity [Fe/H]	[–3.6, 0.3] dex

Table 2 Experimental Results

(a) Experimental results on SDSS stellar spectra					
Estimation Method	Evaluation Method	$\log T_{\rm eff}$ (dex)	$T_{\rm eff}$ (K)	$\log g$ (dex)	[Fe/H] (dex)
The Proposed DNN	MAE	0.0048	64.85	0.1477	0.1129
	ME	0.00005	0.6219	0.0149	0.0043
	SD	0.0075	104.97	0.2180	0.1582
(b) Experimental results evaluated on SDSS stellar spectra summarized from some related literatures					
ANN (Re Fiorentin et al. 2007)	MAE	0.0126	-	0.3644	0.1949
$\rm SVR_G$ (Li et al. 2014)	MAE	0.0075	101.6	0.1896	0.1821
OLS (Tan et al. 2013b)	SD	-	196.5	0.596	0.466
$\rm SVR_l$ (Li et al. 2015)	MAE	0.0060	80.67	0.2225	0.1545
(c) Experimental results on synthetic stellar spectra					
Estimation Method	Evaluation Method	$\log T_{\rm eff}$ (dex)	$T_{\rm eff}$ (K)	$\log g$ (dex)	[Fe/H] (dex)
The Proposed DNN	MAE	0.0011	14.90	0.0182	0.0112
	ME	0.0002	2.861	0.0029	0.0008
	SD	0.0016	22.55	0.0646	0.0153
(d) Experimental results evaluated on synthetic stellar spectra summarized from some related literatures					
ANN (Re Fiorentin et al. 2007)	MAE	0.0030	-	0.0245	0.0269
$\rm SVR_G$ (Li et al. 2014)	MAE	0.0008	-	0.0179	0.0131
OLS (Li et al. 2015)	MAE	0.0022	31.69	0.0337	0.0268



(a) $T_{\rm eff}$ and $\log g$

(b) $T_{\rm eff}$ and [Fe/H]

(c) $\log g$ and [Fe/H]

Fig. 3 Coverage of atmospheric parameters associated with the synthetic spectra.

3.2 Evaluations using Synthetic Spectra

The proposed DNN-based scheme is further tested on 18 969 theoretical star spectra. These spectra are computed using the SPECTRUM software package (v2.76) based on Kurucz's new opacity distribution function (NEWODF; Piskunov et al. 2003) model.

The parameter ranges of these synthetic spectra are listed in Table 1(b) and Figure 3. For effective temperature, these synthetic spectra are computed from 45 parameter values with step 100 K between 4000 K and 7500 K, and 250 K between 7750 K and 9750 K; for metallicity [Fe/H], the spectra are sampled from 27 parameter values with step length 0.2 dex between –3.6 and –1 dex, and 0.1 dex between –1 and 0.3 dex; for surface gravity, these theoretical spectra are sampled on 17 values with step 0.25 dex.

These synthetic spectra are computed with the same wavelength sampling as the real SDSS spectra, and the synthetic spectra are noise-free. In this experiment, the sizes of the training set and test set are 5000 and 13 969 respectively. On this test set, the MAE errors are 14.90 K for effective temperature $T_{\rm eff}$ (0.0011 dex for $\log T_{\rm eff}$), 0.0112 dex for metallicity [Fe/H], and 0.0182 dex for surface gravity $\log g$. More experimental results based on SD and ME are demonstrated in Table 2(c).

3.3 Comparison with Previous Works

Because the estimation of atmospheric parameters from stellar spectra is a fundamental problem in large sky sur-

veys, it has been studied extensively (Re Fiorentin et al. 2007; Jofré et al. 2010; Tan et al. 2013b; Li et al. 2014, 2015).

The atmospheric parameter estimation scheme usually consists of two procedures: representation and mapping. The representation procedure determines how to represent the information contained in a spectrum, for example, Principle Component Analysis (PCA) projections (Jofré et al. 2010; Bu & Pan 2015). The second procedure establishes a mapping from the representation of a spectrum to its parameter to be estimated.

Usually, the two procedures are optimized separately. For example, Re Fiorentin et al. (2007) obtain the representation of a spectrum by a PCA method and parameterize it using an FNN; Li et al. (2015) compute the representation based on a 'Least Absolute Shrinkage and Selection Operator with backward selection' ($\mathrm{LARS}_{\mathrm{bs}}$) method and wavelet analysis, and parameterize the spectrum using a Support Vector Regression method with a linear kernel (SVR_l). Tan et al. (2013b) represent a spectrum using its Lick line index and estimate the atmospheric parameters with an ordinary least squares regression method.

On the contrary, the proposed DNN deals with the spectrum parametrization problem in one unique optimization framework. Some results in the related literature are summarized in Table 2(c) and (d). These demonstrate that the scheme proposed in the present work has excellent performance in stellar spectrum parametrization.

4 CONCLUSIONS

This work investigated the estimation of atmospheric parameters from stellar spectra using deep learning techniques. This parameter estimation problem is commonly referred to as the spectrum-parameterization problem or stellar spectrum classification in related astronomical literatures.

The spectrum-parametrization problem aims to determine a mapping from a stellar spectrum to its atmospheric parameters to be estimated. This work investigated this problem using a DNN. The proposed scheme uses two procedures to determine the mapping: pre-learning and fine-tuning. The pre-learning procedure initializes the structure of the deep network by analyzing the intrinsic properties of a set of empirical data (stellar spectra in this work). A fine-tuning procedure readjusts the network based on specific needs to estimate the atmospheric parameters. Experiments both on real and synthetic spectra show the favorable robustness and accurateness of the proposed scheme.

Acknowledgements We thank the referee for some constructive comments and suggestions. This work is supported by the National Natural Science Foundation of China (NSFC) (Grant Nos. 61273248, 61075033 and 11403056), the Natural Science Foundation of Guangdong Province (2014A030313425 and S2011010003348), the Natural Science Foundation of Shandong Province (ZR2014FM002) and the Joint Research Fund in Astronomy (U1531242) under cooperative agreement between the NSFC and Chinese Academy of Sciences, Guangdong Provincial Engineering Technology Research Center for Data Science.

References

Abazajian, K. N., Adelman-McCarthy, J. K., Agüeros, M. A., et al. 2009, ApJS, 182, 543

Ahn, C. P., Alexandroff, R., Allende Prieto, C., et al. 2012, ApJS, 203, 21

Alam, S., Albareti, F. D., Allende Prieto, C., et al. 2015, ApJS, 219, 12

Allende Prieto, C., Sivarani, T., Beers, T. C., et al. 2008, AJ, 136, 2070

Bahdanau, D., Cho, K., & Bengio, Y. 2014, International Conference on Learning Representations (arXiv:1409.0473)

Bailer-Jones, C. A. L. 2000, A&A, 357, 197

Beers, T. C., Lee, Y., Sivarani, T., et al. 2006, Mem. Soc. Astron. Italiana, 77, 1171

Bu, Y., & Pan, J. 2015, MNRAS, 447, 256

Ciresan, D., Meier, U., Masci, J., & Schmidhuber, J. 2012, Neural Networks, 32, 333

Couprie, C., Farabet, C., Najman, L., & LeCun, Y. 2013, International Conference on Learning Representations (arXiv:1301.3572)

Cui, X.-Q., Zhao, Y.-H., Chu, Y.-Q., et al. 2012, RAA (Research in Astronomy and Astrophysics), 12, 1197

Dahl, G., Mohamed, A.-r., Hinton, G. E., et al. 2010, in Advances in neural information processing systems, 469

Gilmore, G., Randich, S., Asplund, M., et al. 2012, The Messenger, 147, 25

Giridhar, S., Muneer, S., & Goswami, A. 2006, Mem. Soc. Astron. Italiana, 77, 1130

Goodfellow, I. J., Bulatov, Y., Ibarz, J., Arnoud, S., & Shet, V. 2013, International Conference on Learning Representations (arXiv:1312.6082)

Gray, R. O., Corbally, J., C., & Burgasser, A. J. 2009, Stellar Spectral Classification (Princeton: Princeton Univ. Press)

Hinton, G., Deng, L., Yu, D., et al. 2012, IEEE Signal Processing Magazine, 29, 82

Jofré, P., Panter, B., Hansen, C. J., & Weiss, A. 2010, A&A, 517, A57

Krizhevsky, A., Sutskever, I., & Hinton, G. E. 2012, in Advances in Neural Information Processing Systems, 1097

Lee, Y. S., Beers, T. C., Sivarani, T., et al. 2008a, AJ, 136, 2022

Lee, Y. S., Beers, T. C., Sivarani, T., et al. 2008b, AJ, 136, 2050

Lee, Y. S., Beers, T. C., Allende Prieto, C., et al. 2011, AJ, 141, 90

Li, X., Lu, Y., Comte, G., et al. 2015, ApJS, 218, 3

Li, X., Wu, Q. M. J., Luo, A., et al. 2014, ApJ, 790, 105

Luo, A.-L., Zhao, Y.-H., Zhao, G., et al. 2015, RAA (Research in Astronomy and Astrophysics), 15, 1095

Manteiga, M., Ordóñez, D., Dafonte, C., & Arcay, B. 2010, PASP, 122, 608

Ng, A., Ngiam, J., Foo, C. Y., Mai, Y., & Suen, C. 2012, UFLDL Tutorial, *http://deeplearning. stanford.edu/wiki/index.php/UFLDL_Tutorial*

Piskunov, N., Weiss, W. W., & Gray, D. F., eds. 2003, IAU Symposium, 210, Modelling of Stellar Atmospheres

Randich, S., Gilmore, G., & Gaia-ESO Consortium. 2013, The Messenger, 154, 47

Re Fiorentin, P., Bailer-Jones, C. A. L., Lee, Y. S., et al. 2007, A&A, 467, 1373

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. 1986, Nature, 323, 533

Sermanet, P., Kavukcuoglu, K., Chintala, S., & LeCun, Y. 2013, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3626

Smolinski, J. P., Lee, Y. S., Beers, T. C., et al. 2011, AJ, 141, 89

Snider, S., Allende Prieto, C., von Hippel, T., et al. 2001, ApJ, 562, 528

Sutskever, I., Vinyals, O., & Le, Q. V. 2014, in Advances in Neural Information Processing Systems, 3104

Tan, X., Pan, J. C., Wang, J., Luo, A. L., & Tu, L. P. 2013a, Spectroscopy and Spectral Analysis, 33, 1701

Tan, X., Wang, J., Luo, A., et al. 2013b, Spectroscopy and Spectral Analysis, 33, 1397

Willemsen, P. G., Hilker, M., Kayser, A., & Bailer-Jones, C. A. L. 2005, A&A, 436, 379

Yanny, B., Rockosi, C., Newberg, H. J., et al. 2009, AJ, 137, 4377

York, D. G., Adelman, J., Anderson, Jr., J. E., et al. 2000, AJ, 120, 1579

Zhao, G., Chen, Y.-Q., Shi, J.-R., et al. 2006, ChJAA (Chin. J. Astron. Astrophys.), 6, 265