

Consistent and accurate estimation of stellar parameters from HARPS-N Spectroscopy using Deep Learning

Frederik Boe Hüttel*¹ and Line Katrine Harder Clemmensen¹

¹ DTU Compute, Technical University of Denmark Richard Petersens plads 31, 2800 Lyngby, Denmark

Abstract

Consistent and accurate estimation of stellar parameters is of great importance for information retrieval in astrophysical research. The parameters span a wide range from effective temperature to rotational velocity. We propose to estimate the stellar parameters directly from spectral signals coming from the HARPS-N spectrograph pipeline before any spectrum-processing steps are applied to extract the 1D spectrum. We use residual networks and an attention-based model to estimate the stellar parameters. The models estimate both mean and uncertainty of the stellar parameters through the parameters of a Gaussian distribution. The estimated distributions create a basis to generate data-driven Gaussian confidence intervals for the estimated stellar parameters. We show that residual networks and attention-based models can estimate the stellar parameters with high accuracy for low Signal-to-noise ratio (SNR) compared to previous methods. With an observation of the Sun from the HARPS-N spectrograph, we show that the models can estimate stellar parameters from real observational data.

1 Introduction

There exists great variation in the different techniques used to estimate stellar parameters, ranging from decision tree architectures to tailor-made algorithms made for specific astrophysical surveys [14, 15, 22]. Previous research projects that have applied artificial neural networks [1] and deep learning [6] have focused on estimating *effective temper-*

ature (T_{eff}), *surface gravity* ($\log g$), and *metallicity* (Z). Traditionally the stellar parameters are estimated using the extracted 1D stellar spectra from their original CCD spectral images [1, 2, 3, 6]. The methods used for the extraction of the 1D spectrum introduces biases and assumption into the spectrum resulting in biased estimation of stellar parameters, which leads to different research groups obtaining different results when observing the same stars [18]. We argue that one should strive to use an end-to-end deep learning approach, which can estimate appropriate pre-processing steps in the modelling. However the original CCD-spectral images are inherently processed by the spectrograph to make them useable by scientists. The closest thing to an end-to-end approach is using the data from the HARPS-N spectrograph pipeline [17]. In this paper, we propose to use the 2D spectral signal coming from the HARPS-N pipeline to estimate the stellar parameters and present methods for doing so. The main contributions of our approach are:

- The elimination of spectral pre-processing to extract the 1D spectra, as we apply our deep learning models directly to the 2D signal from the HARPS-N pipeline.
- Inclusion of *stellar rotational velocity* ($V \sin i$) estimation and showing predictions accuracy on par with other data-driven methods.
- Quantification of the uncertainty in estimations of the stellar parameters. The estimated distribution provides a basis to create data-driven confidence intervals.
- An attention-based model which attends to the underlying elements of an input spectrum.

*Corresponding Author: fbohy@dtu.dk

2 Data

Previous related research on data-driven stellar parameters estimation has generated synthetic spectra to train and estimate the performance of machine learning models [1, 2, 6]. The use of synthetic spectra provides a unique opportunity to generate a large set of labelled data, as the details of the spectra are known a priori and the SNR can be varied to mimic different telescope exposure times [1, 6]. A drawback of this approach is that the trained weights will be biased towards the physical model generating the data, as there exists a *synthetic gap* between the feature distributions of the synthetic and the observed spectra [6].

2.1 Data generation

We sample the synthetic spectra from a grid of model atmospheres using the ATLAS9 code [12]. The original code is described in detail in [13] and was updated to include new opacity distribution functions as outlined in [4]. The grid was extended by including different rotational velocities ($V \sin i$) using Gray’s methods [7]. The sampling from the grid can be seen in Table 1. We normalise each stellar parameters, so the parameters equally contribute to the loss.

	T_{eff}	$\log g$	Z	$V \sin i$
Min	3500	0	-2.5	0
Max	9750	5	0.5	30
Step	250	0.5	0.5	2
Different values	26	11	7	16

Table 1: Stellar parameters parameters sampled from the grid.

Èchelle orders To generate synthetic spectra images similar to those coming from the HARPS-N pipeline, we split the spectra up into the échelle orders, and stitch them back as an image. We limit the wavelength interval between 5050 to 5350 Å, which corresponds to 8 different échelle orders. We interpolate the wavelengths of the 1D spectrum to match the 2D wavelengths, such that they correspond to the HARPS-N pipeline. We add a linear slant across each order to mimic the observations from the HARPS-N pipeline. We sample the model spectra without any noise, so we can vary the SNR

by adding Gaussian noise during training and testing [6]. An example of the a generated spectra can be seen in Figure 1. The parameters generated are discrete, but we linear interpolate between samples to create observations with continuous parameters [6]. The final dimensional of the spectra are 8 rows with 4096 pixels in each.

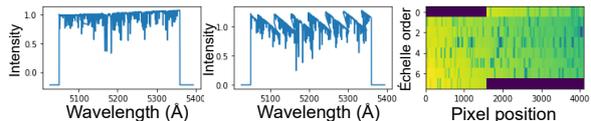


Figure 1: This figure highlights the different approaches. *Left*: A sample of a 1D model spectrum, the methods for generation these are presented in subsection 2.1. *Middle*: A sample of a 1D model spectrum similar to the HARPS-N pipeline, by inclusion of the échelle orders as described in section 2.1. *Right*: The spectral image of the spectrum in the middle.

3 Method

This section describes how we capture heteroscedastic uncertainty in the estimated stellar parameters, present the attention-based architecture and provide a framework to denoise samples with low SNR.

3.1 Heteroscedastic Uncertainty Estimation

Heteroscedastic regression assumes that the uncertainty of observations vary with input \mathbf{x} [10]. This uncertainty in the observations can be quantified by the distribution $p(y|\mathbf{x})$, where the expected value is considered the best estimate of the parameters, and the variance of the distribution describes the uncertainty [19]. We learn the distribution using parameters θ and parameterize the learnt distribution as a Gaussian.

$$p(y|\mathbf{x}, \theta) = \mathcal{N}(\mu_{\theta}(\mathbf{x}), \Sigma_{\theta}(\mathbf{x})) \quad (1)$$

Where $\mu_{\theta}(\mathbf{x})$ is a vector of size 4, and $\Sigma_{\theta}(\mathbf{x})$ aims to learn the covariance of the 4 parameters. In order to estimate the parameters in the function in Equation 1, we minimise the negative log-likelihood [10].

$$\mathcal{L}(\theta) = \frac{N}{2} \log |\Sigma_{\theta}(\mathbf{x})| + \frac{1}{2} \sum_{i=1}^N (y_i - \mu_{\theta}(\mathbf{x}_i))^T \Sigma_{\theta}(\mathbf{x})^{-1} (y_i - \mu_{\theta}(\mathbf{x}_i)) \quad (2)$$

3.2 Attention-based model

The soft-attention model used in this work is inspired by the architecture from [9], in combination with the attention blocks presented in [20, 21]. We construct an attention architecture which uses any number of intermediate feature maps x_n from a convolutional neural network in combination with a global feature map g (The last layer in the convolutional neural network), to compute an attention map in attention blocks [9]. The attention map $\alpha_n \in [0, 1]$ is used to identify salient features in the input as the output of an attention block is the element-wise multiplication of the input feature-map and the attention map: $\hat{x}_n = \alpha_n \cdot x_n$. We denote the channels of a given feature map F_x and F_g as the channels from the global feature map. F_{int} is the channels of the convolutional weights in the attention block to ensure g and x_n have the same channels. Formally we can compute the attention map α_n as follows.

$$q_{att}^l = \psi^T \left(\sigma_1 \left(\mathbf{W}_x^T \mathbf{x}_n + \mathbf{W}_g^T \mathbf{g} + \mathbf{b}_{xg} \right) \right) + b_{\psi} \\ \alpha_n = \sigma_2 \left(q_{att}^l \left(\mathbf{x}_n, \mathbf{g}; \Theta_{att} \right) \right) \quad (3)$$

where $\sigma_1(x)$ is the activation function of the neural network, and $\sigma_2(x)$ is the *softmax* operation, such that the attention map sums to one [20]. The set of parameters Θ_{att} contains the convolutional weights $\mathbf{W}_x \in \mathbb{R}^{F_x \times F_{int}}$ and $\mathbf{W}_g \in \mathbb{R}^{F_g \times F_{int}}$, which are used to linearly transform the input tensors using a channel-wise $1 \times 1 \times 1$ convolution. The weights $\psi \in \mathbb{R}^{F_{int} \times 1}$ combines the features from all the channels into a 1-channel attention map. In addition we also include a bias term $b_{\psi} \in \mathbb{R}^{F_{int}}$ in Θ_{att} . The parameters of the attention block (the convolutional layers) are trained using standard back-propagation [20]. Each attention block outputs $\mu_{\theta}(\mathbf{x})$ and $\Sigma_{\theta}(\mathbf{x})$

Aggregation strategy To ensure that all the attention maps α_n learn meaningful features, we aver-

age the outputs of the attention blocks. The overall attention architecture is presented in Figure 2.

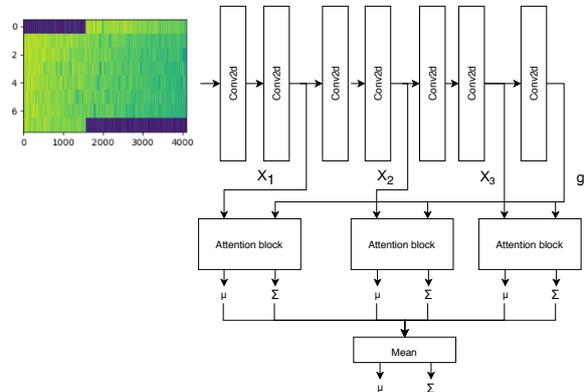


Figure 2: Figure showing the overall architecture of the attention network used in this work. This figure does not reflect the trained model and is only for overview purposes. Illustration inspired by Figure 2 in [9].

We train an attention network with three attention blocks and 11 convolutional layers. After each attention block, we apply 2×2 max pooling. The pooling, ensures that the later feature maps contains more local features. The initial block will find global features of the input, and subsequent blocks will attend to more local features [20]. We compare the attention model to a residual network consisting of 5 residual blocks [8] and three fully connected layers. We limit the convolutional kernel sizes be (1×7) and add zero padding, which limits the artefacts generated by the convolutional kernels to only be within the spectral image.

3.3 Denoising Auto-encoder

In practice, the obtained SNR of observations is often lower than expected, which can result in an unsatisfactory performance on the stellar label estimation. We propose to use a *Denoising auto-encoder* (DAE) [23] to remove noise and ensure that samples with low SNR can be used in stellar label estimation. We assume the noise of a sample \mathbf{x} is equal to Gaussian white noise across the entire spectrum and denote the corrupted sample \mathbf{x}' , which is a fair assumption given Gaussian noise is used to vary the SNR of a sample [6]. We learn ϕ and θ , by optimising the encoder $f_{\phi}(\mathbf{x}')$ and the de-

SNR	$\Sigma_{\theta}(\mathbf{x})$	Model	T_{eff}	$\log(g)$	Z	$V \sin i$
20	Diagonal	Residual-network	76.9	0.138	0.055	0.71
20	Diagonal	Attention-network	73.0	0.135	0.053	0.69
20	Diagonal	DAE Residual-network	72.3	0.133	0.052	0.67
20	Diagonal	DAE Attention-network	70.9	0.134	0.049	0.57
20	Full	Residual-network	83.8	0.143	0.060	0.75
20	Full	Attention-network	79.2	0.146	0.055	0.72
20	Full	DAE Residual-network	89.1	0.150	0.060	0.72
20	Full	DAE Attention-network	72.9	0.137	0.049	0.58
200	-	Cannon2 [3]	46.8	0.066	0.036	-
200	-	StarNet [6]	31.2	0.053	0.025	-
100	Diagonal	Residual-network [†]	19.5	0.053	0.026	0.30
100	Diagonal	Attention-network [†]	<i>12.9</i>	<i>0.045</i>	<i>0.013</i>	<i>0.15</i>

Table 2: Mean-Absolute error based on the mean prediction from models. Using the DAE we can remove noise and increase the performance of our models. [†] models are trained on a limited data-set to match the parameter ranges presented in previous related work [6]. The limited data-set only contains T_{eff} between 4000K and 6000K, while the other parameters stay the same.

coder $g_{\theta}(\mathbf{z})$, where \mathbf{z} is the latent code extracted by f_{ϕ} , by minimising the reconstitution error between the corrupted sampled \mathbf{x}' and the original sample \mathbf{x} [23].

$$\mathcal{L}(\theta, \phi) = \|\mathbf{x} - g_{\theta}(f_{\phi}(\mathbf{x}'))\|^2 \quad (4)$$

4 Experiments and results

The results are obtained on a test set of 5667 synthetic model spectra. Gaussian noise is added to match the SNR of the training set. We trained the attention-network and residual network with a diagonal covariance matrix and a full covariance matrix. The attention and residual network are optimised using the Adam [11] variant of stochastic gradient (SGD) descent with a learning rate of 0.0001. The models are trained for 750 epochs. The DAE is trained for 1500 epochs using the AdamW [16] variant of SGD with a learning rate of 0.0003. For all models the learning rate is decayed by a γ of 0.1 for the last 50 epochs

4.1 Experimental Results

In Table 2, we present the model performance on the Mean-Absolute error (*MAE*) on the 4 stellar parameters. The MAE obtained in this work is similar to the results obtained by other related work [3, 6], however, we have eliminated the need

for extraction of the 1D spectra. The reader should be aware of the fact that the results obtained by related work are achieved from a different data-pipeline. The DAE can remove noise from samples and have a positive impact on model performance when the SNR is low (Table 2). The incorporation of the DAE makes the model robust to noisy observations. This suggest that a DAE might also improve performance for traditional methods, which typically do not work well when data exhibits low SNR. When estimating a full covariance matrix, the model performance is slightly lower compared to learning a diagonal covariance (Table 2). Based on the results presented in Table 2, We will continue to show the results from the attention-model using a diagonal covariance matrix with a auto-encoder (DAE Attention-network). We evaluate the uncertainty with Gaussian confidence intervals, using the diagonal elements in $\Sigma_{\theta}(\mathbf{x})$.

Residuals The residuals appear to be equally distributed around 0 (Figure 3). We detect a little bias in some of the estimation. The model appears to underestimates the temperature for really high temperatures stars. The variance within each stellar parameters does not appear to be constant across the entire domain. For low values of Z , the models estimates have high variance across all the four labels, it appears that for spectra with low Z , there is high variance in the residuals (Figure 3).

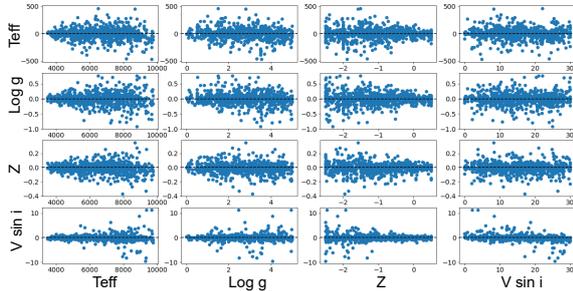


Figure 3: The residuals across the different stellar parameters. *Right to left ordering:* T_{eff} , $\log g$, Z and $V \sin i$. *Top to bottom ordering:* T_{eff} , $\log g$, Z and $V \sin i$

Standard deviations The estimated standard deviations are obtained from the diagonal elements of $\Sigma_{\theta}(\mathbf{x})$. The models estimate high uncertainty for high temperature stars (Figure 4) which is expected as the stellar features are sparse for hot stars [6]. The same pattern for low metallicity stars (Figure 4), where the estimated uncertainty is higher for low values, compared to high values of metallicity, which support our assumption of heteroscedastic variance.

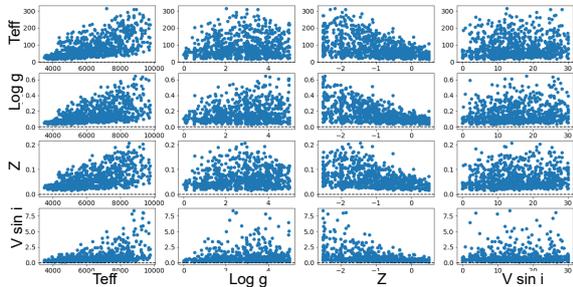


Figure 4: The estimated standard deviation from the residuals presented in Figure 3 as a function of the true parameters.

Uncertainty estimation We assess the quality of the estimated standard deviations by evaluation of residuals within 1 or 2 standard deviations.

The estimated distributions approximate the Gaussian theoretical values (Table 3). One might argue that the residual networks are overestimating the variance of the learned distribution. We argue this happens because we train the residual networks with dropout in the fully connected layers, so the model is trained with uncertainty in \mathbf{x} .

Model	$\epsilon < \Sigma_{\theta}(\mathbf{x})$	$\epsilon < 2\Sigma_{\theta}(\mathbf{x})$
Gaussian	68.2%	95.1%
Residual-network	79.9%	98.4%
Attention-network	65.5%	93.4%

Table 3: Table showing percentages of observations that are within $\mu \pm \Sigma_{\theta}(\mathbf{x})$ and $\mu \pm 2\Sigma_{\theta}(\mathbf{x})$. The models are trained with $\text{SNR} \approx 20$.

4.2 Test on HARPS-N observation

In order to evaluate the synthetic gap, we assess the models on an observation of the Sun coming from the HARPS-N spectrograph. Due to the extreme apparent brightness of the Sun, these observations obtain high SNR of ≈ 200 [5]. All real observations contain telluric lines (absorption lines coming from the Earth's atmosphere), which is not present in the synthetic data.

Model	T_{eff}	$\log(g)$	Z	$V \sin i$
HARPS-N	5750	4.44	0	2
Residual-net	5791.6 ± 140.1	4.72 ± 0.28	0.035 ± 0.15	0.762 ± 1.76
Attention-net	5325.2 ± 10.0	2.15 ± 0.04	-0.576 ± 0.01	5.226 ± 0.40

Table 4: Estimated values for the HARPS-N observation. Confidence bands are estimated using Gaussian confidence intervals. Models trained with $\text{SNR} \approx 100$.

The residual network can estimate the parameters of the Sun, and use the estimated uncertainty to setup confidence intervals (Table 4).

4.3 Visual evaluation of attention

Visual evaluation of the estimated attention feature maps α shows that the model attends to some of the composite elements in the spectra (Figure 5). Magnesium b have spectral lines at 5172 Å and is often used by traditional methods when estimating the stellar parameters. Based on the high activation of the attention feature map α at this absorption line, the attention-network attends to this element in the spectrum. The intermediate feature maps are up-sampled after the pooling layers, to fit the input spectral image, this leads to attended features outside the signal.

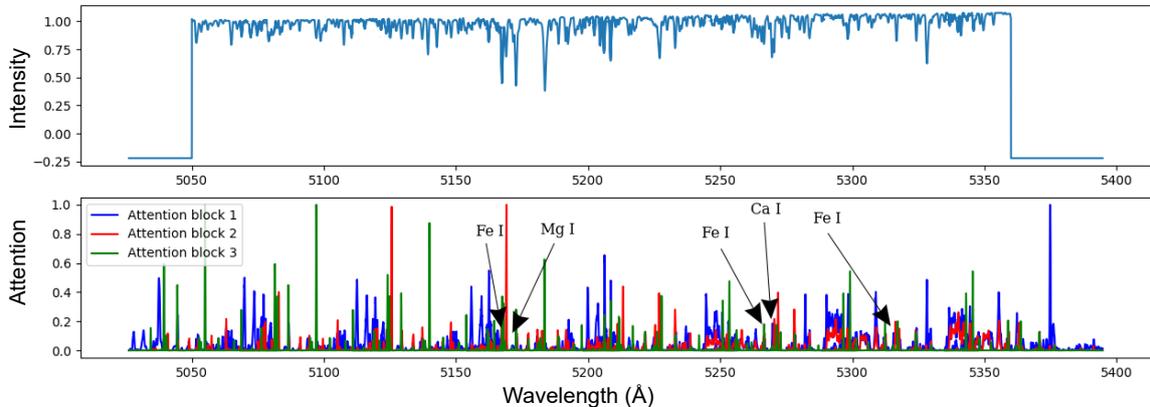


Figure 5: *Top*: Input spectrum, showing the intensity of each wavelength. *Bottom*: Visualisation of the 3 different attention feature maps α from the 3 different attention blocks used in this work. Elements observed in the range highlighted.

5 Conclusion

We have focused on a data-driven estimation of stellar parameters based on the spectral signal directly from the HARPS-N pipeline. Based on the results obtained by using denoising models, the use of such models should be applicable not only in a deep learning approach but also for more traditional physics-based methods which typically underperform with data that exhibits low SNR. The estimation of a multivariate Gaussian also lays the groundwork for future research to explore full Bayesian approaches such as Markov chain Monte Carlo or variational inference methods. The attention models provide a way to reason about the importance of the different composite elements of a spectrum, as the models presented here attend to some of the salient underlying elements in a spectrum. Since the telluric lines are absent from our data-set future work could include these to reduce the synthetic gap. The physical knowledge required to analyse the spectra holds tremendous value, and it is as essential as the estimations. We, therefore, encourage future research to continue the path towards an end-to-end deep learning method, while acknowledging the importance of the physical composition of the underlying spectra.

Acknowledgements We wish to thank Lars A. Buchhave and Alexander Dybdahl Ratchke from the National Space Institute at the Technical Uni-

versity of Denmark, who has been the inspiration for this work and provided the data-set.

Earlier work An early version of this work was presented at the workshop on Machine Learning and the Physical Sciences at NeurIPS 2020.

References

- [1] C. A. L. Bailer-Jones. Stellar parameters from very low resolution spectra and medium band filters te , $\log g$ and (m/h) using neural networks. 2000.
- [2] C. A. L. Bailer-Jones, M. Irwin, G. Gilmore, and T. von Hippel. Physical parametrization of stellar spectra: the neural network approach. *Monthly Notices of the Royal Astronomical Society*, 292(1):157–166, 11 1997.
- [3] A. Casey, D. W. Hogg, M. K. Ness, H.-W. Rix, A. Y. Q. Ho, and G. F. Gilmore. The cannon 2: A data-driven model of stellar spectra for detailed chemical abundance analyses. 2016.
- [4] F. Castelli and R. L. Kurucz. New grids of atlas9 model atmospheres. *arXiv preprint astro-ph/0405087*, 2004.
- [5] X. Dumusque, A. Glenday, D. F. Phillips, N. Buchschacher, A. C. Cameron, M. Ceconi, D. Charbonneau, R. Cosentino, A. Ghedina, D. W. Latham, and et al. Harps-n observes

- the sun as a star. *The Astrophysical Journal*, 814(2):L21, Nov 2015.
- [6] S. Fabbro, K. Venn, T. O’Brian, S. Bialek, C. KIELTY, F. Jahandar, and S. Monty. An application of deep neural networks in the analysis of stellar spectra, 2017.
- [7] D. F. Gray. *The observation and analysis of stellar photospheres*. Cambridge University Press, 2005.
- [8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.
- [9] S. Jetley, N. A. Lord, N. Lee, and P. H. S. Torr. Learn to pay attention. *CoRR*, abs/1804.02391, 2018.
- [10] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? *CoRR*, abs/1703.04977, 2017.
- [11] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2014.
- [12] R. Kurucz and B. Bell. Kurucz cd-rom 13. *ATLAS9 stellar atmosphere programs and*, 2:1, 1993.
- [13] R. L. Kurucz. Atlas: A computer program for calculating model stellar atmospheres. *SAO Special report*, 309, 1970.
- [14] Y. Lee, T. Beers, J. Carlin, H. Newberg, Y. Hou, G. Li, A.-L. Luo, Y. Wu, M. Yang, H. Zhang, W. Zhang, and Y. Z. NIAOT. Application of the segue stellar parameter pipeline to lamost stellar spectra. *The Astronomical Journal*, 150:187, 12 2015.
- [15] Y. Lee, T. Beers, T. Sivarani, C. Prieto, L. Koesterke, R. Wilhelm, P. Re Fiorentin, C. Bailer-Jones, J. Norris, C. Rockosi, B. Yanny, H. Newberg, K. Covey, H. Zhang, and A.-L. Luo. The segue stellar parameter pipeline. i. description and comparison of individual methods. *The Astronomical Journal*, 136:2022, 10 2008.
- [16] I. Loshchilov and F. Hutter. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101, 2017.
- [17] M. Mayor, F. Pepe, D. Queloz, F. Bouchy, G. Rupprecht, G. Lo Curto, G. Avila, W. Benz, J. L. Bertaux, X. Bonfils, T. Dall, H. Dekker, B. Delabre, W. Eckert, M. Fleury, A. Gilliotte, D. Gojak, J. C. Guzman, D. Kohler, J. L. Lizon, A. Longinotti, C. Lovis, D. Megevand, L. Pasquini, J. Reyes, J. P. Sivan, D. Sosnowska, R. Soto, S. Udry, A. van Kesteren, L. Weber, and U. Weilenmann. Setting New Standards with HARPS. *The Messenger*, 114:20–24, Dec. 2003.
- [18] M. Ness, D. W. Hogg, H.-W. Rix, A. Y. Q. Ho, and G. Zasowski. The cannon: A data-driven approach to stellar label determination. *The Astrophysical Journal*, 808(1):16, Jul 2015.
- [19] D. A. Nix and A. S. Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN’94)*, volume 1, pages 55–60 vol.1, 1994.
- [20] O. Oktay, J. Schlemper, L. L. Folgoc, M. C. H. Lee, M. P. Heinrich, K. Misawa, K. Mori, S. G. McDonagh, N. Y. Hammerla, B. Kainz, B. Glocker, and D. Rueckert. Attention u-net: Learning where to look for the pancreas. *CoRR*, abs/1804.03999, 2018.
- [21] J. Schlemper, O. Oktay, L. Chen, J. Matthew, C. L. Knight, B. Kainz, B. Glocker, and D. Rueckert. Attention-gated networks for improving ultrasound scan plane detection. *CoRR*, abs/1804.05338, 2018.
- [22] R. Smiljanic, A. J. Korn, M. Bergemann, A. Frasca, L. Magrini, T. Masseron, E. Pancino, G. Ruchti, I. San Roman, L. Sbordone, and et al. Thegaia-eso survey: The analysis of high-resolution uves spectra of fgk-type stars. *Astronomy & Astrophysics*, 570:A122, Oct 2014.
- [23] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning, ICML ’08*, page 1096–1103, New York, NY, USA, 2008. Association for Computing Machinery.