





Convolutional Neural Networks and Stokes Response Functions

REBECCA CENTENO ¹, NATASHA FLYER ², LIPI MUKHERJEE,¹ RICKY EGELAND ^{1,*}, ROBERTO CASINI ¹,
TANAUSÚ DEL PINO ALEMÁN,^{3,4} AND MATTHIAS REMPEL¹

¹*High Altitude Observatory, National Center for Atmospheric Research, Boulder, CO*

²*Flyer Research LLC, Boulder, CO*

³*Instituto de Astrofísica de Canarias, E-38205, La Laguna, Tenerife, Spain*

⁴*Departamento de Astrofísica, Facultad de Física, Universidad de La Laguna, Tenerife, Spain*

ABSTRACT

In this work, we study the information content learned by a convolutional neural network (CNN) when trained to carry out the inverse mapping between a database of synthetic Ca II intensity spectra and the vertical stratification of the temperature of the atmospheres used to generate such spectra. In particular, we evaluate the ability of the neural network to extract information about the sensitivity of the spectral line to temperature as a function of height. By training the CNN on sufficiently narrow wavelength intervals across the Ca II spectral profiles, we find that the error in the temperature prediction shows an inverse relationship to the response function of the spectral line to temperature, this is, different regions of the spectrum yield a better temperature prediction at their expected regions of formation. This work shows that the function that the CNN learns during the training process contains a physically-meaningful mapping between wavelength and atmospheric height.

Keywords: Stokes polarimetry — Chromosphere — Radiative Transfer — Machine Learning

1. INTRODUCTION

In the rarefied environment of the solar chromosphere the radiation field is weakly coupled to the local thermodynamic conditions, and common approximations used to solve the polarized radiative transfer problem in the photosphere break down. The two main ones are 1) the assumption of local thermodynamical equilibrium (LTE), and 2) that the source of spectral line polarization is only due to the Zeeman effect. A more generalized approach needs to be adopted in the chromosphere, recognizing that the majority of spectral lines generated there form under non-LTE conditions, and in the presence of anisotropic excitation mechanisms that lead to the production of scattering polarization. As a result, the forward modeling of chromospheric spectral lines can be computationally very expensive, because of the need to solve the very complex problem of the statistical equilibrium of the atom under such conditions, and under the joint action of the Hanle and Zeeman effects.

The possibility to exploit spectropolarimetric observations of the solar atmosphere to infer its physical (thermodynamic and magnetic) state depends on our ability to understand and constrain a very complex and generally ill-posed inverse problem. Spectral-line inversion codes for the photosphere have been developed and successfully employed for several decades thanks to the simplifications that are possible in the forward modeling of the formation of photospheric spectral lines. These codes carry out an “inverse mapping” from the observations to the physical state of the Sun’s atmosphere. A traditional solution approach relies on non-linear least squares methods in order to minimize a merit function that quantifies the differences between the observed and modeled spectra. This entails solving the forward modeling of the spectral line many times until the inversion converges. In the case of chromospheric modeling, this approach adds yet another layer of iterations on top of an already computationally demanding problem. For this reason, spectral line inversion codes for chromospheric lines typically adopt different approximations to make

Corresponding author: Rebecca Centeno
rce@ucar.edu

* now at NASA Johnson Space Center, Houston, TX.

the inversion problem tractable, by either relying on simplified atmospheres (Asensio Ramos et al. 2008) or assuming that the spectral line polarization is only due to the Zeeman effect (Socas-Navarro et al. 2015; de la Cruz Rodríguez & van Noort 2017; Milić & van Noort 2018; de la Cruz Rodríguez et al. 2019). Even after these simplifications, non-LTE inversion codes are still computationally demanding, have finicky initialization procedures, and often present unstable convergence behaviors, requiring a high level of experience to run them.

A different approach to spectral line inversions, which is based on machine learning (ML), is becoming very popular nowadays. While this approach is not completely new (see e.g. Carroll & Staude 2001; Socas-Navarro 2005), it is gaining more traction in recent times because of the growing access to large computing power resources and to open-source ML software packages.

By training a convolutional neural network (CNN) on the inversion results (and inversion fits) of synthetic spectra, Milić & Gafeira (2020) were able to speed up the spectral line inversion of these data by 5 orders of magnitude, compared to the performance of a traditional Levenberg-Marquardt-based inversion scheme. Following a slightly different philosophy, Asensio Ramos & Díaz Baso (2019) trained two different CNN architectures on synthetic spectropolarimetric data from radiative magneto-hydrodynamical MURaM simulations. Instead of doing this on a pixel-by-pixel basis, these authors fed the CNNs 2-dimensional patches of the simulated Sun’s surface, taking advantage of the horizontal continuity of the data. The CNNs so constructed are able to retrieve the thermodynamical and magnetic properties of the photosphere as a function of geometric height, as well as to deconvolve the data from the instrument’s point spread function. This technique was successfully applied to Hinode/SP observations and the authors predict that it could invert all the data from the entire mission in just a matter of minutes. A recent work by Higgins et al. (2021) shows that ML techniques can also be used to cross-calibrate spectral line inversion results from instruments that observe the Sun in fundamentally different ways (in their case, Hinode/SP and SDO/HMI). Sainz Dalda et al. (2019) apply deep learning techniques to the inversion of chromospheric intensity spectra from the IRIS mission, resulting in a decrease of 5 to 6 orders of magnitude in the computing time for the retrieval of the thermodynamical properties of the chromosphere. While the accuracy of the results is often reported to be good in a qualitative manner, Gafeira et al. (2021) claim that these kinds of approaches can be used to find suitable guess atmospheres to feed to traditional non-linear least squares inversion codes, resulting in a significant reduction in the number of inversion cycles, and cutting the computing time by a factor of 2 to 4.

An obvious limitation of the NN approach to spectropolarimetric inversions is that the atmospheric and magnetic field models that can be inferred through data inversion are restricted to those contained in the physical parameter space adopted for the NN training (i.e., NNs cannot be used to extrapolate solutions outside of the training set). Contrary to how traditional inversion codes work, the NN inversion does not call an underlying radiative transfer model. Given enough data, however, it is able to reconstruct a function that carries out an approximate mapping between the Stokes profiles at the atmospheric properties that lead to them. This begs the question of whether or not this mapping function is able to glean physically-meaningful information from the databases on which it has been trained.

In this work, we conduct a detailed study of the information content that a CNN is able to infer when trained to carry out the inverse mapping between a database of synthetic intensity spectra and the atmospheric temperature profiles used to generate them. In particular, we analyze what the CNN is able to learn about the sensitivity to temperature of the two lines of Ca II at 3934 Å (K-line) and 8542 Å (IR2), as a function of wavelength and atmospheric height. This sensitivity is derived from the errors in the temperature prediction when the CNN is trained to carry out the inverse mapping on narrow spectral windows within each of the spectral lines. In Section 2, we describe the numerical setups used to create the databases of atmospheric models and Ca II spectra. Section 3 presents the CNN architectures used for this work, and describes the methodology that we follow to quantify the information content learned by the CNN. A comparison of these results with the response functions of Ca II K and IR2 to temperature is presented in Section 4, followed by our concluding remarks and an outlook for future developments.

2. CREATION OF DATABASES OF ATMOSPHERIC MODELS AND SPECTRA

In order to compile a sufficiently large database of 1-dimensional (1D) model atmospheres and spectra on which to train the CNN, we used a snapshot of a 3D radiative magneto-hydrodynamic simulation from the MURaM code. In particular, we chose a Quiet Sun (QS) simulation (Rempel 2014), with a horizontal spatial domain of 24.5x24.5 Mm² and a vertical extent of 8.2 Mm (~2 Mm of which are above the solar surface). The spatial grid is sampled with a 16 km spacing in all directions. The simulation was started from thermally relaxed, non-magnetic convection; successively, a

constant, vertical, seed magnetic field of 10^{-3} G was added, unevenly distributed over the horizontal plane. Once the simulation reached the saturation regime the average magnetic field strength was 60 G, while the average net vertical flux was negligible.

While this particular MURaM simulation does not produce realistic physical conditions in the solar chromosphere, the main goal of this work is to understand the information content learned by the CNN from its training dataset. The methodology presented here can later be utilized on increasingly realistic numerical simulations, with the ultimate objective of developing a functional ML spectral line inversion code that can be applied to solar observations.

The database of 1D model atmospheres was compiled by extracting individual columns from the 3D MURaM cube. In order avoid redundancy and correlation in our atmospheric database, we selected columns on a sparse horizontal grid (keeping one in every 8, in each spatial direction).

In order to generate Stokes spectra from the different realizations of the MURaM atmospheres we use the Hanle-RT code (del Pino Alemán et al. 2016). Hanle-RT is a polarized radiative transfer code for 1-D (plane-parallel) atmospheres. It accepts any multi-term atom (LS-coupling scheme) without hyperfine structure,¹ and in the presence of magnetic fields of arbitrary strength. The code takes into account the effects of non-isotropic radiative excitation of the atomic system, which are responsible for the manifestation of scattering polarization. Therefore, the code can model seamlessly all possible regimes of magnetic-induced effects on the polarization of spectral lines: the Hanle, Zeeman, and (incomplete) Paschen-Back effects, as well as the polarization effects associated with level crossing and anti-crossing interference. The code fully implements the effects of inelastic, elastic, and depolarizing collisions, and also includes a comprehensive description of partially coherent scattering (partial redistribution, or PRD) from polarized atoms (limited to 3-term atoms of the Λ -type; see Casini & Manso Sainz 2016), which is important for modeling deep chromospheric lines.

The atomic model of Ca II used to solve the statistical equilibrium (SE) includes the five bound levels of the lowest three terms of the ion, as well as the Ca III ground state. Such a model describes five bound-bound radiative transitions, namely, the Ca II H & K lines in the near UV and the Ca II IR triplet. The IR triplet is computed assuming complete frequency redistribution (CRD), whilst the H & K lines are treated in PRD. Cross-distribution (Raman scattering) is allowed between these sets of transitions. For computational expediency, the system is treated as a 5-level system, rather than the actual 3-term atom. Thus, the effects of J -level quantum interference (incomplete Paschen-Back effect, and level crossing and anti-crossing) are not taken into account in this test model, but they would have a negligible impact on the Stokes I profile anyway.

The MURaM simulation provides the thermodynamical and magnetic quantities as a function of geometric height needed to compute the Stokes spectra: temperature, gas pressure, line-of-sight (LOS) velocity, and magnetic field vector. Often times, microturbulent velocity is added to the models for broadening purposes, in order to increase the similarity between synthetic spectra and observations (de la Cruz Rodríguez et al. 2012). A direct comparison with observations, however, is out of the scope of this work, so we do not see the need to artificially increase the complexity of the model for this sake.

Before feeding these quantities to Hanle-RT, we set the LOS velocity to 0 km/s in the interest of preserving the (non-linear) mapping between absolute wavelength and atmospheric height. If velocities are present, this mapping is subject to further smearing due to the Doppler shifting of the spectral lines². The 1D thermodynamical profiles are then smoothed in the vertical direction by applying a Savitzky-Golay filter (Savitzky & Golay 1964) with a 1st-order polynomial and a window size of 17 grid points (\equiv 256 km). Then, they are re-sampled onto a 32 km grid, and the portion of the model that lies below -192 km is discarded (a portion of atmosphere below 0 km is retained³ in order to maintain its small contribution to the formation of the wings of the spectral lines). This vertical smoothing was carried out in order to reduce the LOS gradients and the inherent fluctuations in the model atmosphere, while retaining the main trends and the variability across the different realizations in the atmospheric database. This process yields 36,828 unique 1D model atmospheres, that span from -192 km to 1888 km in height, sampled in steps of 32 km. These model atmospheres are given to Hanle-RT as an input. Hanle-RT then calculates the Stokes profiles for the 5 Ca II lines contained in the model atom, assuming an observing geometry of $\mu = 1.0$ (where μ is the cosine of the heliocentric angle). Figure 1 shows an example of a Ca II K intensity profile as a function of wavelength. The wavelength sampling

¹ Multi-level atoms with hyperfine structure can also be modeled with this code.

² The reader interested in the accuracy of the temperature prediction using CNNs when velocities are included in the spectral line synthesis, is referred to section B of the Appendix.

³ height = 0 km is defined as the point where the mean continuum optical depth at 500 nm in the MURaM cube is unity ($\tau_{500} = 1$).

is non-equidistant, being much denser in the core of the line than in the wings (the former is enlarged in the inset of the figure for better viewing). This is necessary in order to ensure numerical stability while solving the radiative transfer because of the steeper gradients in the core of the line compared to the wings. The blue and green boxes delimit narrow (5 grid-point) wavelength windows, the purpose of which will be discussed in the next section.

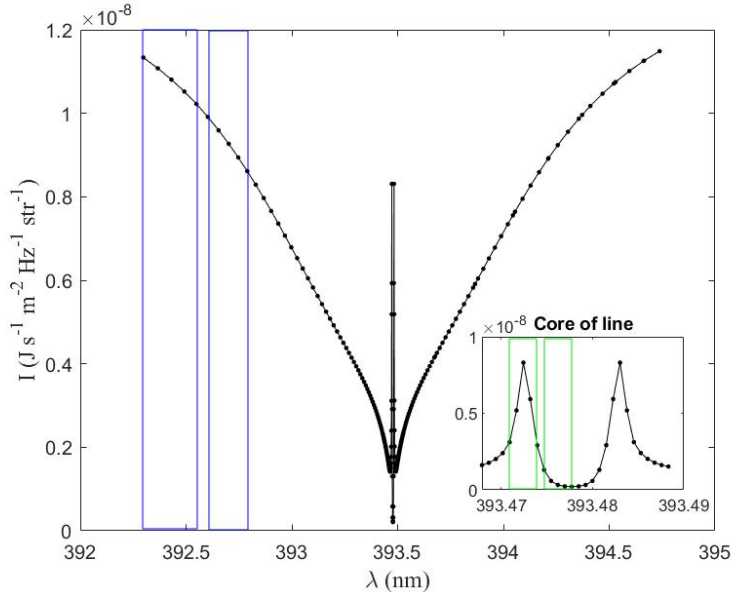


Figure 1. An example of a Ca II K Stokes I profile as a function of wavelength. The black dots represent the wavelength sampling across the line, which is non-equidistant. The figure inset zooms into the core of the line. The vertical blue and green boxes showcase a sliding wavelength window, covering five wavelength sampling points, used to analyze the sensitivity of the spectral line to temperature as a function of height (described in Section 3) .

In this way, we create two databases, one of model atmospheres and one of Stokes spectra, with a known one-to-one correspondence between each individual 1D physical state and each Stokes vector. We can then use these databases to train a neural network to learn the mapping between the synthetic “observations” (spectra) and the physical conditions in the Sun’s atmosphere. In what follows, we will map the Stokes I profiles of the Ca II K and IR2 lines (separately) to the atmospheric temperature as a function of height using the regression CNN outlined in the next section. The distribution of temperature as function of height for the 36,828 samples in our atmospheric database is shown in Figure 2. In the photosphere (~ 0 -500 km), the temperature samples are concentrated in a narrow range of values (4000 K to 6500 K), while as we approach the chromosphere, the distribution spreads, showing a much larger range of temperatures (typically between 2000 K and 7000 K). As we will see later, this will affect the accuracy of the CNN temperature retrieval, with the lowest errors found below 500 km.

3. CNN ARCHITECTURE

The regression CNN developed for this study is programmed in Keras (Chollet et al. 2015) (see Appendix for the code). As shown in Table 1, it is a 9-layer CNN with 3 convolutional layers, each followed by an Exponential Linear Unit (ELU) non-linear activation function with its hyperparameter set to 1. The choice of activation function relies on the faster learning speeds and higher accuracy that the ELU confers to deep NNs (Clevert et al. 2016). A mean absolute percentage error ($100/N \sum_{i=1}^N |true_i - predicted_i|/|true_i|$) loss function is used to calculate the model error from the 60 unit output layer (corresponding to the temperature sampled at 60 geometric heights) during the optimization process. The optimization algorithm used is Adam (Kingma & Ba 2017), a stochastic first-order gradient method with the learning rate set to 0.01. The batch size is 50, with 67% of the data used for training, 3% for validation, and 30% for testing. The network’s batch size was kept optimally small to achieve higher accuracy without capturing noise (as illustrated in Keskar et al. 2017). The CNN was trained for 150 epochs. A min-max scaling was used on the input data (Stokes I) to normalize it between 0 and 1 (Pedregosa et al. 2011).

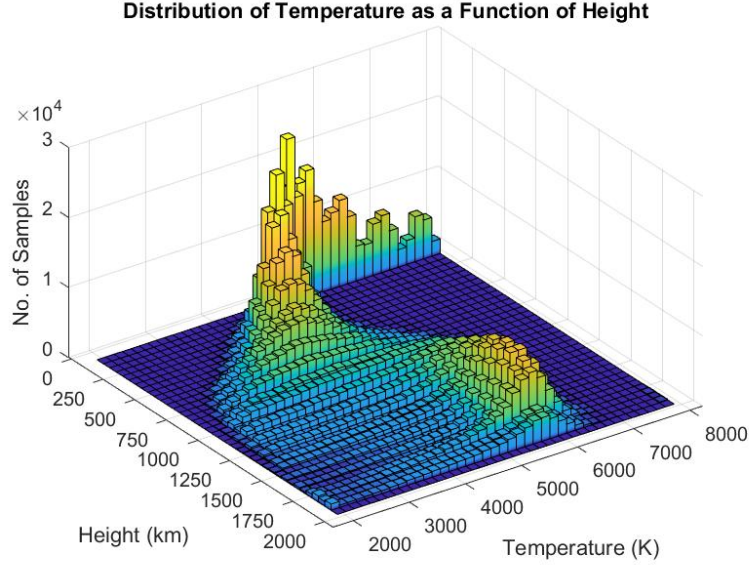


Figure 2. 3D histogram of the distribution of temperature as a function of height for the 36,828 samples in the database.

Layer Type	Description	Activations/ Output Shape	Learnable Parameters
image_input	253×1 image (Whole Spectral Line)		—
Convolution	32 3×1 convolutions, stride [1 1]	253×1×32	Weights : $(3 \times 1) \times 32$ Bias : 32×1
ELu	ELU nonlinear activation function	—	—
Convolution	64 3×1 convolutions, stride [1 1]	253×1×64	Weights : $(3 \times 32) \times 64$ Bias : 64×1
ELu	ELU nonlinear activation function	—	—
Convolution	64 3×1 convolutions, stride [3 1]	85×1×64	Weights : $(3 \times 64) \times 64$ Bias : 64×1
ELu	ELU nonlinear activation function	—	—
Dense	60 unit fully connected output layer (Temperature output at 60 heights)	60×1	Weights : $(85 \times 64) \times 60$ Bias : 60×1
Loss_MAPE	Mean absolute percentage error	—	—

Table 1. CNN architecture for training the Ca II K spectral line (253 values of the intensity) to map the temperature at 60 geometric heights.

Of special note to this CNN architecture is that no pooling layers are used to reduce the spatial size of the data. Such layers usually occur in CNN architectures to downsample the feature maps that are inputted into the next convolutional layer and thus hopefully retain the most important features in a map while decreasing the computation in the network. In general, this is usually warranted when attempting to do classification on high-dimensional images (e.g. reducing an 128 by 128 image to a vector of 10 for classification through a softmax function). However, in the presented problem we are doing 1D convolutions on a vector of length 253 (wavelength samples) with an output of length 60 (geometric heights), in an attempt to map the spectral wavelength domain to the temperature-height

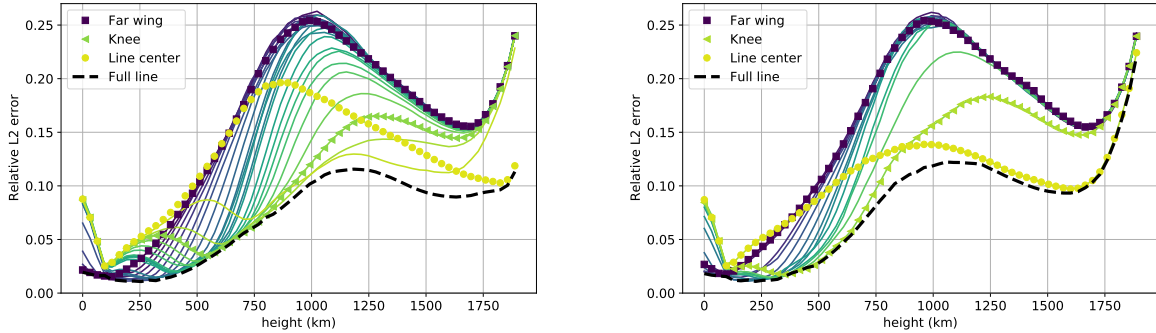


Figure 3. CNN-derived sensitivity of the spectral line. The results for Ca II K are shown on the left and those for Ca II 8542 Å are shown on the right. The colored lines show the L2-error in the temperature prediction when the CNN is trained on short spectral windows along the spectral line. Dark blue/purple corresponds to the case when the window is in the far wing of the line and lightest shade of green corresponds to its placement in the core. The black dashed line corresponds to the error in the temperature prediction when training the CNN on the whole spectral line at once.

domain. Instead of pooling layers, we used a *striding* approach to perform downsampling in the last convolution layer. By increasing the stride length of the convolutional filter we can still reduce the spatial size of the output (layer output = (input - filter size)/stride + 1) while the convolutional filter weights benefit from “*learning*” on a larger input feature map than would have occurred if a pooling layer had been used instead.

This CNN architecture is trained on the Ca II K and IR2 intensity spectra separately, to carry out the inverse mapping from the spectral profiles to the atmospheric temperature as a function of height. The black dashed lines in Fig. 3 show the CNN prediction uncertainty, as a function of height, for the the CNN trained on Ca II K (left) and IR2 (right). The errors are given in terms of the L2-norm:

$$\text{L2} = \sqrt{\frac{\sum_{i=1}^N (t_i - p_i)^2}{\sum_{i=1}^N t_i^2}} \quad (1)$$

where t and p represent the true and predicted temperatures, respectively, and the summation is carried out over all the samples in the database. The L2-norm is very similar to the root mean square error, but the normalization is done with respect to the true vector as opposed to the number of samples, N . The advantage is that it gives a relative error that is dimensionless.

The results for the Ca II K and IR2 lines are strikingly similar. As hinted earlier, the error in the temperature prediction is smaller in the lower parts of the atmosphere (for both spectral lines) than it is at higher layers. Interestingly, the error in the temperature prediction at the very top of the atmosphere (above ~ 1700 km) is much larger when the CNN is trained on IR2 than when it is trained on the K-line. This is likely due to the fact that the core of the latter forms higher in the atmosphere (see, for instance [Leenaarts et al. 2013](#)), and therefore retains sensitivity for mapping the temperature at these heights. An analysis of (and comparison with) the Stokes response function to temperature for both of these lines is presented in section 4. Figure 4 shows examples of the true (blue) vs. predicted (red) temperature stratification for a random selection of samples in the testing set. While the CNN is not able to capture the small-scale variations in the temperature, it predicts its overall stratification with strikingly good accuracy.

3.1. CNN setup for mapping spectral line sensitivity to temperature at different heights

Given enough data, the CNN-based inversion is able to reconstruct a function that carries out an approximate mapping between the spectral line intensity and the temperature stratification. Throughout the remainder of this manuscript we will try to quantify how much information this CNN inversion method is able to glean about the sensitivity of different parts of the spectral line to different heights in the atmosphere, and whether or not this bears any relation to the temperature response function of the spectral line. In other words, we are interested in finding out if there is physically-meaningful information in the mapping function between the wavelength and the height domains that the CNN learned during its training.

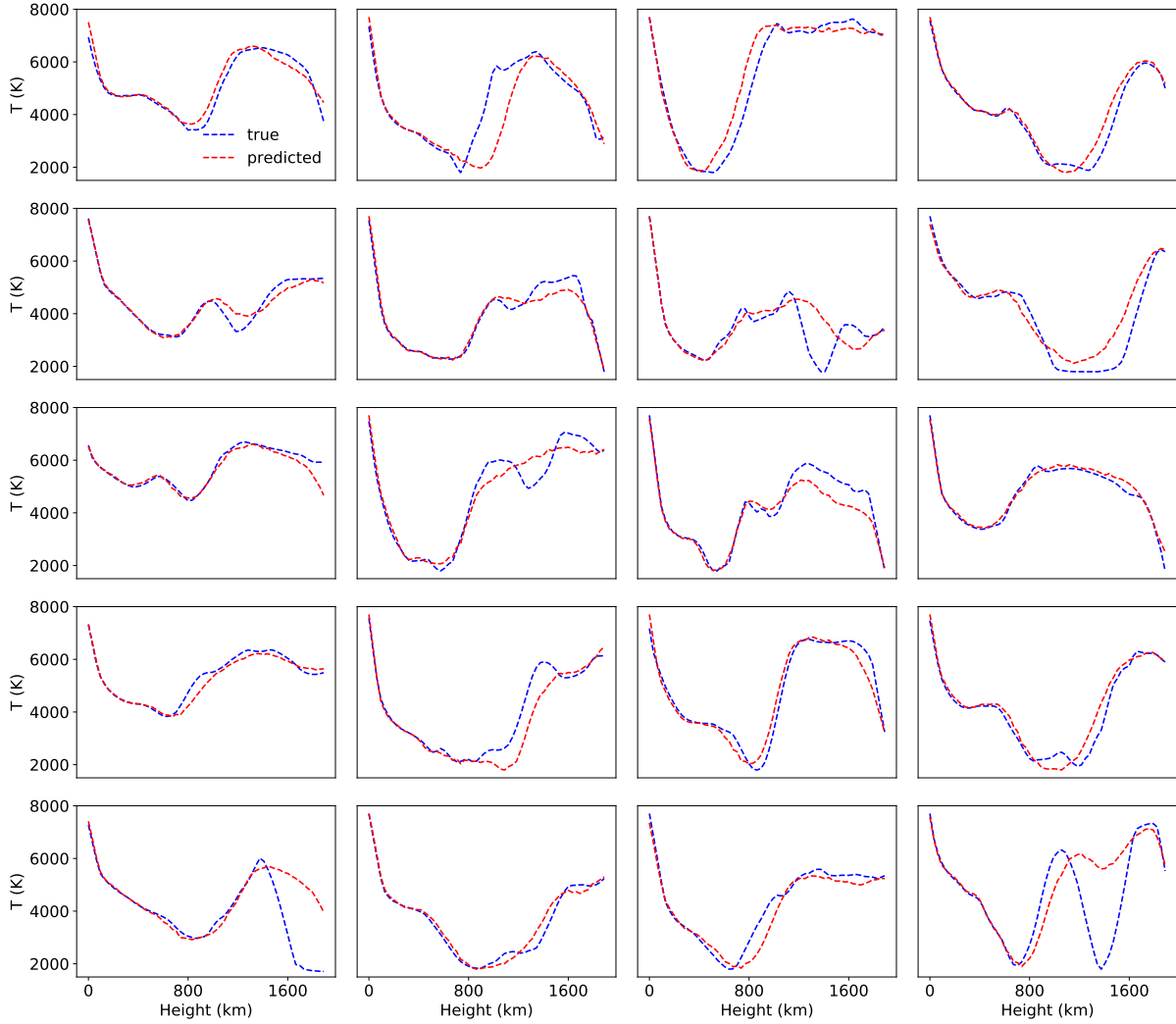


Figure 4. True (blue) vs. predicted (red) temperature as a function of height for a selection of 25 random samples from the testing set.

To analyze how different wavelengths of the Ca II K and IR2 spectral lines are sensitive to the temperature at different atmospheric heights, we train a two-layer CNN (given in Table 2) on a short wavelength window composed of five consecutive wavelength samples in the wing of Stokes I (such a wavelength window is represented by the left-most blue box in Fig. 1). The window, always of five wavelength samples, is then slid across the spectral line with a stride of 5 wavelength positions and the CNN is trained again on the new wavelength range (indicated by the second blue box in Fig. 1). This process is repeated all the way to the central wavelength of the spectral line (right-most green box in the inset of Fig. 1), and the L2 errors for the temperature prediction as a function of height are computed for each position of the short spectral window. Only half of the spectral line is considered because of the (near perfect) symmetry properties of Stokes I in the absence of velocity gradients. Notice that the window width (in wavelength units) narrows as it moves from the wing to the core of the line due to the non-equidistant wavelength sampling. We decided to work with this uneven sampling because the information density in the core of the line is much larger than in the wings.

The colored lines in Fig. 3 show the L2 prediction errors derived from training the two-layer CNN on these short spectral windows, for Ca II K (left) and for IR2 (right). The hue of the lines in the plots reflects the positioning of the 5-sample wavelength windows along the spectral line, with dark blue corresponding to the far wing and light green to the line core (the results for specific wavelength windows are highlighted by different symbols in the plots).

Layer Type	Description	Activations/ Output Shape	Learnable Parameters
image_input	5×1 image (Window of Spectral Line)		—
Convolution	32 3×1 convolutions, stride [1 1]	5×1×32	Weights : $(3 \times 1) \times 32$ Bias : 32×1
ELu	ELU nonlinear activation function	—	—
Convolution	64 3×1 convolutions, stride [1 1]	5×1×64	Weights : $(3 \times 1) \times 64$ Bias : 64×1
ELu	ELU nonlinear activation function	—	—
Dense	60 unit fully connected output layer (Temperature output at 60 heights)	60×1	Weights : $(5 \times 32) \times 60$ Bias : 60×1
Loss_MAPE	Mean absolute percentage error	—	—

Table 2. Two-layer CNN for training on a short wavelength window within the spectral line to carry out the mapping between the intensity and the temperature as a function of height.

Examine the results for the K-line (left). At the lowest heights (< 160 km), the best prediction is achieved by the darkest blue line (also represented with squares), which corresponds to the errors of the CNN trained in the far wing of the spectral line (more than 1.2 nm away from the line center, λ_0). As the spectral window is slid inwards (and the color of the lines in the plot becomes a lighter shade of blue), the temperature prediction improves for the mid and high photospheric layers (200-500) km. The window that includes the “knee” of the spectral line (represented with triangles) yields the best temperature predictions just above the photosphere-chromosphere interface. Here, the “knee” of the spectral line is taken to be the apparent wing-core boundary, marked by the vertical orange lines in Fig. 6 (which for Ca II K lies at $\lambda - \lambda_0 \sim 0.011$ nm). The “knee” of the spectral line seems to mark the separation between two differentiated regimes in terms of the height of maximum sensitivity (HMS) of the spectral line as a function of wavelength. Above 750 km, the best predictions are attained by the three curves with the lightest shade of green, which include the wavelength ranges in the core of the line proper (from $\sim \lambda_0 - 0.01$ nm all the way to the line center). It seems as if this wavelength range contained the vast majority of the information pertaining to the mid-to-upper chromosphere. The line with the solid circles (which corresponds to the spectral window that includes λ_0) intersects all other lines and delivers the lowest error of all at the top of the atmosphere. These results are qualitatively compatible with what we know about spectral line formation: the wings of the line form in the photosphere while the core forms in the high chromosphere (see e.g. Bjørgen et al. 2018).

The results for Ca II 8542 Å (right) are qualitatively similar, with the core of the line yielding better predictions at higher atmospheric layers and the wing of the line performing better at lower heights. The main difference lies in the prediction error at the very top of the atmosphere (above ~ 1700 km), which is significantly worse for IR2 than for the K-line. This is likely due to the fact that the latter forms higher in the atmosphere than the former.

The dashed line represents the errors derived from training the three-layer CNN on the whole spectral line at once. Not surprisingly, it gives the lower bound on the error of the temperature prediction across all spectral windows, and in some height ranges, it performs better than any/all of the individual windows. It is possible that no single narrow wavelength interval contains enough information to capture the LOS integration due to the photon mean-free-path. Also, note that the CNN used to train the whole spectral line uses a larger input and is deeper than the two-layer CNN used to train on short spectral windows, and thus it may have a larger learning ability, leading to slightly smaller errors overall.

The L2-error metric appears to be inversely related to the sensitivity of the spectral line to temperature as a function of height across all of the short wavelength intervals. In the following section we will compare this CNN-derived sensitivity to the numerical response function of the spectral lines to temperature.

4. NUMERICAL RESPONSE FUNCTIONS TO TEMPERATURE

All spectral lines form over a range of heights in the Sun’s atmosphere. Response functions (RF; see, e.g. del Toro Iniesta & Ruiz Cobo 2016) are defined as the change in the intensity and polarization of a spectral line in response to

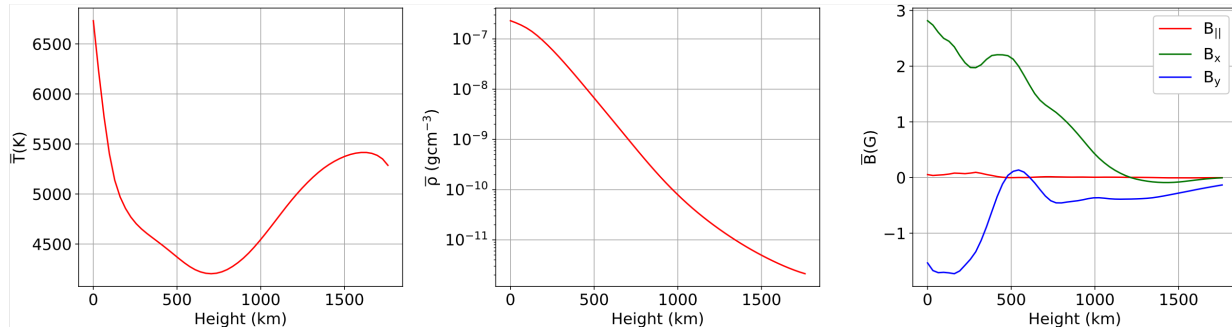


Figure 5. Physical properties of the spatially averaged MURaM quiet Sun run.

a small perturbation of an atmospheric parameter in a narrow range of optical depths. RFs provide information about the sensitivity of the Stokes profiles to the atmospheric properties as a function of height, and can help us determine whether a spectral line carries or not (useful) information to diagnose a particular physical parameter at such height.

In this section, we calculate the response functions of Ca II K and IR2 to temperature following the strategy described in [Quintero Noda et al. \(2016\)](#). Instead of working on an optical depth grid, we work directly on a geometric height grid in order to compare the RFs to the CNN-derived sensitivity obtained for each spectral line in the previous section.

RFs are model dependent ([del Toro Iniesta & Ruiz Cobo 2016](#)). This means that each 1D model atmosphere extracted from the MURaM simulation will result in a slightly different response function to temperature. In order to produce meaningful RFs for the whole dataset, we computed the average atmosphere from the entire MURaM atmospheric database and used that as the reference model for the calculation. Fig. 5 shows the horizontally averaged atmospheric quantities as a function of height. The left panel represents the average temperature (\bar{T}), which drops fast from the bottom of the photosphere towards the temperature minimum (around 700 km), and then rises into the chromosphere. Note that this particular MURaM simulation does not achieve realistic temperatures at the base of the corona, which has an impact on the heights of formation of the spectral lines. The middle panel shows the average density ($\bar{\rho}$), which drops exponentially with height; and lastly, the panel on the right contains the averages of the 3 magnetic field components (B_{\parallel} , B_x and B_y , where \parallel refers to the vertical direction, and x and y are in the perpendicular plane). This quiet-Sun MURaM simulation is set up to have a zero vertical net flux ($\bar{B}_{\parallel} = 0$) and small, yet fluctuating averages of the horizontal components. In this figure, \bar{B}_{\parallel} is not strictly zero. This is likely due to the fact that our MURaM dataset comprises just over $\sim 1\%$ of the original snapshot (only 1 in every 8 grid points in each horizontal direction was used to construct the atmospheric database).

For each height z_i , we apply a perturbation of $\Delta T = +25$ K to the temperature of the base model and calculate the emergent spectra I_i^+ . We repeat this exercise by applying a negative perturbation of $\Delta T = -25$ K to the base model and computing the corresponding I_i^- . From the emergent intensity spectra, the response of Stokes I to temperature for each wavelength and height is given by:

$$\mathbf{R}_T(\lambda, z_i) = \frac{I_i^+ - I_i^-}{2I_C \Delta T} \quad (2)$$

where $\mathbf{R}_T(\lambda, z_i)$ is the response function, normalized by the local continuum intensity I_C .

Fig. 6 shows the response functions of Ca II K (left) and IR2 (right) to temperature as a function of wavelength on the x-axis and height on the y-axis. A positive response (teal) means that the intensity increases in response to a positive temperature perturbation. As expected, the wings of both spectral lines are sensitive to changes in temperature at photospheric heights, with a very slow dependence on wavelength. It is not until we reach the knee of the spectral line (marked with an orange vertical line), that a sharp increase in the HMS begins. This change in behavior takes place around $\sim 500 - 700$ km for both lines. It is interesting to note that, in this particular atmospheric model, the sensitivity of the core of the K-line to temperature extends to the very top (and beyond) of the model atmosphere, while the sensitivity of the IR2 line drops off around ~ 1700 km. This is qualitatively and quantitatively compatible with the results presented in the previous section, where the L2 error for the CNN trained on IR2 shows a sharp increase above this height. In this context, the L2-error metric of the CNN performance shows an *inverse relationship* to the response function of the spectral line to temperature.

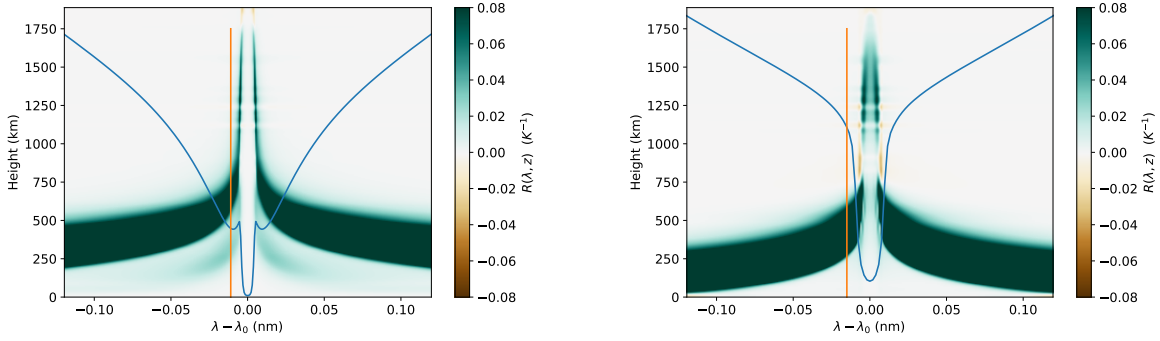


Figure 6. Response functions of Ca II K (left) and Ca II 8542 Å (right) to temperature in the average MURaM atmosphere. The x-axis represents wavelength and the y-axis geometric height in the mean atmospheric model. A positive response (teal) means that the intensity increases in response to a positive temperature perturbation. The average Stokes I (in blue) is overplotted for reference. The vertical orange line marks the approximate position of the “knee” of the spectral lines.

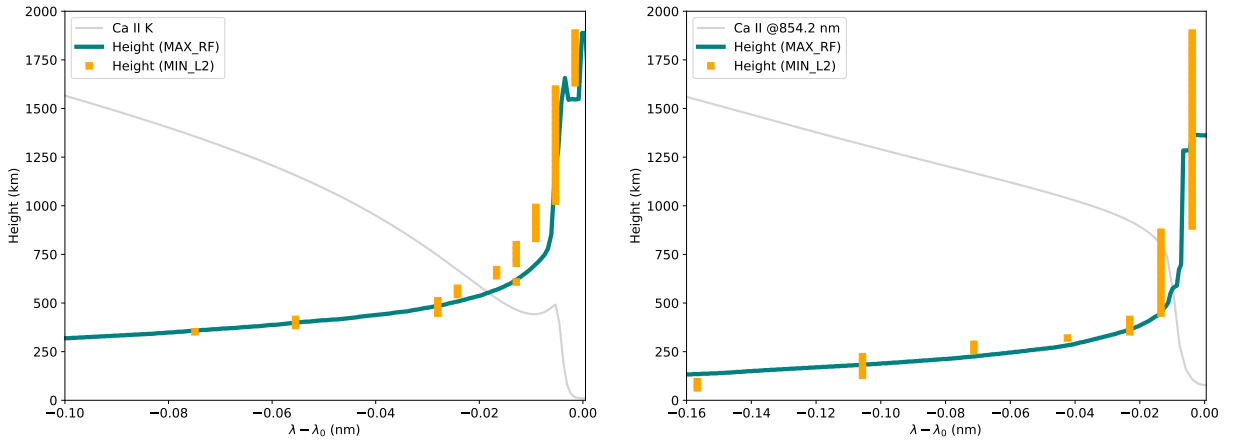


Figure 7. Quantitative comparison between the spectral line response function and the L2-error metric of the CNN temperature prediction (Ca II K on the left and IR2 on the right). In teal we show the wavelength that yields the maximum value of the RF at each height. The orange squares show which wavelength window (represented by its central wavelength) delivers the lowest L2-error for the temperature prediction at each height. The average Stokes I (in grey) is over-plotted for reference.

Figure 7 shows a quantitative comparison between these two metrics of spectral line sensitivity. These plots show, in teal, the wavelength that delivers the maximum value of the RF for any given height, or in other words, the height at which the RF peaks as a function of wavelength. We can extract a similar relationship from the L2 curves presented in Fig. 3: at each height in the atmosphere we find the L2 curve (and the central λ of the corresponding wavelength interval) that delivers the lowest error in the temperature prediction. This is shown by the orange squares. Note that the orange curve shows a lot more degeneracy than the one derived from the RF, owing to the size of the wavelength intervals on which the CNN was trained. For this reason, a single wavelength interval can yield the lowest L2-error for a range of heights. However, the curves follow each other very well, and seem to tell the same story: the HMS shows a monotonic rise as we move from the far wing to the knee of the spectral line; at this wavelength, a dramatic shift in behavior takes place, leading to a sudden increase of this quantity.

The agreement between the information contained in the L2-error metric and the RF of the spectral line indicates that the non-linear function learned by the CNN contains physically-meaningful information about the mapping between spectral wavelength and atmospheric height.

5. CONCLUSIONS

In this work, we analyze the information content learned by a neural network when trained to carry out the inverse mapping between the intensity spectra of two Ca II lines (separately) and the atmospheric temperature as a function of height. The numerical setup comprised 36,828 1D model atmospheres extracted from a radiative magneto-hydrodynamic MURaM simulation, and an equal number of Ca II K and IR2 Stokes I spectra computed with the Hanle-RT code.

First, a three-layer CNN was trained to map the whole spectral line to the atmospheric temperature as a function of height, and its performance was evaluated using an L2-error metric to capture the uncertainty in the temperature prediction (dashed lines in Fig. 3). As expected, this prediction error is partially a reflection of the spread of the temperature at each height in the model, but it also captures information about the sensitivity of the spectral line to temperature at different atmospheric heights. This is epitomized by the fact that IR2 leads to much larger prediction errors than the K-line above 1700km, which speaks to the average formation heights of the cores of these two spectral lines in this particular MURaM model.

Then, a two-layer CNN was trained on short wavelength intervals that were slid across the spectral line, from the far wing to the line core. For each spectral window, the CNN is trained to retrieve the temperature as a function of height, and the L2-error metric is collected in each case (the latter is shown by the colored lines in Fig. 3). The performance of the CNN trained on these short spectral windows shows a monotonic trend in the height that delivers the lowest prediction error. For the lowest atmospheric heights, the window positioned in the far wing of the spectral line delivers the best result (i.e. the smallest error). As we progressively move to higher atmospheric layers, the lowest errors are delivered by spectral windows that are increasingly closer to the core of the line. The best prediction at the top of the atmosphere is achieved only for the window that contains the central wavelength, λ_0 .

Lastly, we compute the numerical response functions to the temperature in order to evaluate the sensitivity of the two Ca II lines to thermal perturbations in the model. The response function gives us a quantitative measure of the change in Stokes I as a function of wavelength when the temperature is perturbed at any given height in the atmosphere. We can draw a one-to-one correspondence between the L2-error metric of the CNN performance (trained on short wavelength intervals) with the response function of the spectral line to temperature. By comparing the wavelength dependence of the height of maximum sensitivity of the RF and the height of minimum L2-error of the CNN temperature prediction, we see a one-to-one mapping between both curves.

This work shows that a NN trained to carry out the inverse mapping between Stokes I profiles and the atmospheric temperature used to generate them, is able to glean information about the wavelength-height sensitivity of the spectral line in a physically meaningful way.

- 1 This material is based upon work supported by the National Center for Atmospheric Research, which is a major
- 2 facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977. We would like
- 3 to acknowledge high-performance computing support from Cheyenne (doi:10.5065/D6RX99HX) provided by NCAR's
- 4 Computational and Information Systems Laboratory, sponsored by the National Science Foundation.

REFERENCES

- Asensio Ramos, A., & Díaz Baso, C. J. 2019, *A&A*, 626, A102, doi: [10.1051/0004-6361/201935628](https://doi.org/10.1051/0004-6361/201935628)
- Asensio Ramos, A., Trujillo Bueno, J., & Landi Degl'Innocenti, E. 2008, *ApJ*, 683, 542, doi: [10.1086/589433](https://doi.org/10.1086/589433)
- Bjørgen, J. P., Sukhorukov, A. V., Leenaarts, J., et al. 2018, *A&A*, 611, A62, doi: [10.1051/0004-6361/201731926](https://doi.org/10.1051/0004-6361/201731926)
- Carroll, T. A., & Staude, J. 2001, *A&A*, 378, 316, doi: [10.1051/0004-6361:20011167](https://doi.org/10.1051/0004-6361:20011167)
- Casini, R., & Manso Sainz, R. 2016, *ApJ*, 824, 135, doi: [10.3847/0004-637X/824/2/135](https://doi.org/10.3847/0004-637X/824/2/135)
- Chollet, F., et al. 2015, *Keras*, GitHub. <https://github.com/fchollet/keras>
- Clevert, D.-A., Unterthiner, T., & Hochreiter, S. 2016, *Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)*. <https://arxiv.org/abs/1511.07289>
- de la Cruz Rodríguez, J., Leenaarts, J., Danilovic, S., & Uitenbroek, H. 2019, *A&A*, 623, A74, doi: [10.1051/0004-6361/201834464](https://doi.org/10.1051/0004-6361/201834464)
- de la Cruz Rodríguez, J., Socas-Navarro, H., Carlsson, M., & Leenaarts, J. 2012, *A&A*, 543, A34, doi: [10.1051/0004-6361/201218825](https://doi.org/10.1051/0004-6361/201218825)
- de la Cruz Rodríguez, J., & van Noort, M. 2017, *SSRv*, 210, 109, doi: [10.1007/s11214-016-0294-8](https://doi.org/10.1007/s11214-016-0294-8)
- del Pino Alemán, T., Casini, R., & Manso Sainz, R. 2016, *ApJL*, 830, L24, doi: [10.3847/2041-8205/830/2/L24](https://doi.org/10.3847/2041-8205/830/2/L24)

- del Toro Iniesta, J. C., & Ruiz Cobo, B. 2016, *Living Reviews in Solar Physics*, 13, 4, doi: [10.1007/s41116-016-0005-2](https://doi.org/10.1007/s41116-016-0005-2)
- Gafeira, R., Orozco Suárez, D., Milić, I., et al. 2021, arXiv e-prints, arXiv:2103.09651. <https://arxiv.org/abs/2103.09651>
- Higgins, R. E. L., Fouhey, D. F., Antiochos, S. K., et al. 2021, arXiv e-prints, arXiv:2108.12421. <https://arxiv.org/abs/2108.12421>
- Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., & Tang, P. T. P. 2017, On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima. <https://arxiv.org/abs/1609.04836>
- Kingma, D., & Ba, J. 2017, Adam: A Method for Stochastic Optimization. <https://arxiv.org/abs/1412.6980>
- Leenaarts, J., Pereira, T. M. D., Carlsson, M., Uitenbroek, H., & De Pontieu, B. 2013, *ApJ*, 772, 90, doi: [10.1088/0004-637X/772/2/90](https://doi.org/10.1088/0004-637X/772/2/90)
- Milić, I., & Gafeira, R. 2020, *A&A*, 644, A129, doi: [10.1051/0004-6361/201936537](https://doi.org/10.1051/0004-6361/201936537)
- Milić, I., & van Noort, M. 2018, *A&A*, 617, A24, doi: [10.1051/0004-6361/201833382](https://doi.org/10.1051/0004-6361/201833382)
- Pedregosa, F., Varoquaux, G., Gramfort, A., et al. 2011, *Journal of Machine Learning Research*, 12, 2825
- Quintero Noda, C., Shimizu, T., de la Cruz Rodríguez, J., et al. 2016, *MNRAS*, 459, 3363, doi: [10.1093/mnras/stw867](https://doi.org/10.1093/mnras/stw867)
- Rempel, M. 2014, *ApJ*, 789, 132, doi: [10.1088/0004-637X/789/2/132](https://doi.org/10.1088/0004-637X/789/2/132)
- Sainz Dalda, A., de la Cruz Rodríguez, J., De Pontieu, B., & Gošić, M. 2019, *ApJL*, 875, L18, doi: [10.3847/2041-8213/ab15d9](https://doi.org/10.3847/2041-8213/ab15d9)
- Savitzky, A., & Golay, M. J. E. 1964, *Analytical Chemistry*, 36, 1627
- Socas-Navarro, H. 2005, *ApJ*, 621, 545, doi: [10.1086/427431](https://doi.org/10.1086/427431)
- Socas-Navarro, H., de la Cruz Rodríguez, J., Asensio Ramos, A., Trujillo Bueno, J., & Ruiz Cobo, B. 2015, *A&A*, 577, A7, doi: [10.1051/0004-6361/201424860](https://doi.org/10.1051/0004-6361/201424860)

APPENDIX

A. CNN ARCHITECTURE

Below is the Keras CNN model trained to carry out the mapping between Stokes I and the atmospheric temperature as a function of geometric height. Note that the 2-layer CNN used to train on short spectral intervals has a very similar architecture but is missing the last convolution layer.

```
import numpy as np
import tensorflow as tf
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from tensorflow.keras.layers import Conv2D
from keras.models import Sequential
from keras.layers import Dense, Flatten, Activation
from keras.optimizers import Adam
from sklearn.model_selection import train_test_split

scaler = MinMaxScaler()
% X is the input vector, Stokes I and
% Y is the output vector which is the temperature at corresponding geometric heights
scaler.fit(X)
X = scaler.transform(X)

Xx = X.reshape(X.shape[0], X.shape[1], 1, 1)
Yy = Y.reshape(Y.shape[0], Y.shape[1], 1)
n_inputs, n_outputs = X.shape[1], Y.shape[1],

model = Sequential()
model.add(Conv2D(32, (3,1), strides=(1, 1),activation="elu",padding='same', input_shape=(n_inputs,1,1)))
model.add(Conv2D(64, (3,1), strides=(1, 1),activation="elu",padding='same'))
model.add(Conv2D(64, (3,1), strides=(3, 1),activation="elu",padding='same'))
model.add(Flatten())
model.add(Dense(n_outputs))
opt=Adam(lr=0.01, epsilon=None, decay=0.0, amsgrad=False)
model.compile(loss="mape", optimizer=opt)
batch_size = 50
xtr, xte, ytr, yte=train_test_split(Xx, Yy, test_size=0.30, random_state=42)
model.fit(xtr, ytr, batch_size=batch_size,validation_split=0.03,epochs=150)
```

B. CNN PERFORMANCE IN THE PRESENCE OF LOS VELOCITIES

The inversion of spectral lines generated in static atmospheres is of limited practical value. Below we show preliminary results in the accuracy of the temperature retrieval when the Ca II K synthetic spectra are computed in the presence of LOS velocities from the original MURaM simulation. Figure 8 shows the relative L2 temperature prediction error in the presence of LOS velocities (blue), which should be compared to the case without velocities (black; this is identical to the black dashed line in the left panel of Fig. 3). The two curves present error values within 5% of each other up to ~ 1600 km, height above which the L2-error metric for the case with velocities shoots up. This is work in progress and we believe there is room for improvement.

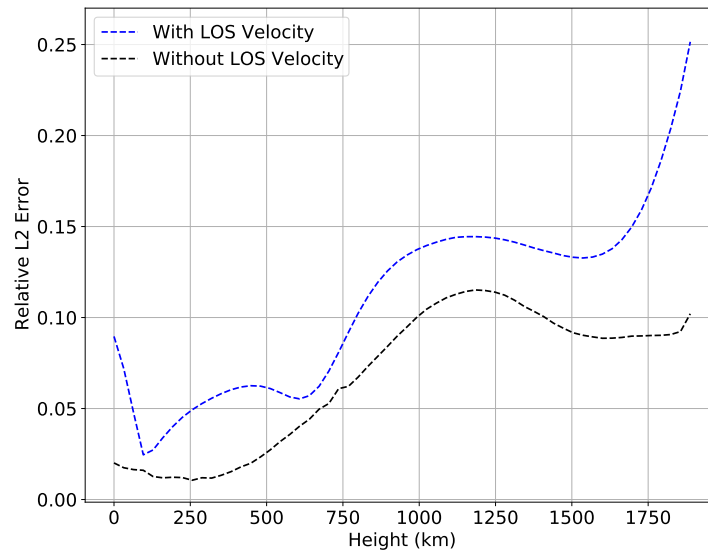


Figure 8. Relative L2 error of the temperature retrieval from the CNN inversion of Ca II K, with and without LOS velocities.