The Astrophysical Journal, 994:110 (20pp), 2025 November 20 https://doi.org/10.3847/1538-4357/ae0cbc

Machine Learning Inference of Stellar Properties Using Integrated Photometric and Spectroscopic Data

Ilay Kamai[1], Alex M. Bronstein[2,3], and Hagai B. Perets[1,4]

[1] Physics Department, Technion-Israel Institute of Technology, Haifa 32000, Israel; ilay.kamai@campus.technion.ac.il
[2] Computer Science Department, Technion-Israel Institute of Technology, Haifa 32000, Israel
[3] Institute of Science and Technology Austria, Klosterneuburg 3400, Austria
[4] ACRO, Open University of Israel, R'anana, Israel

Received 2025 July 15; revised 2025 September 25; accepted 2025 September 25; published 2025 November 19

Abstract

Stellar astrophysics relies on diverse observational modalities—primarily photometric light curves and spectroscopic data—from which fundamental stellar properties are inferred. While machine learning (ML) has advanced analysis within individual modalities, the complementary information encoded across modalities remains largely underexploited. We present the dual embedding for stellar astronomy (DESA) model, a novel multimodal foundation model that integrates light curves and spectra to learn a unified, physically meaningful latent space for stars. DESA first trains separate modality-specific encoders using a hybrid supervised/self-supervised scheme, and then aligns them through DualFormer, a transformer-based cross-modal integration module tailored for astrophysical data. DualFormer combines cross- and self-attention, a novel dual-projection alignment loss, and a projection-space eigendecomposition that yields physically structured embeddings. We demonstrate that DESA significantly outperforms leading unimodal and self-supervised baselines across a range of tasks. In zero- and few-shot settings, DESA's learned representations recover stellar color–magnitude and Hertzsprung–Russell diagrams with high fidelity ($R^2 = 0.92$ for photometric regressions). In full fine-tuning, DESA achieves state-of-the-art accuracy for binary star detection (AUC = 0.99, AP = 1.00) and stellar age prediction (RMSE = 0.94 Gyr). As a compelling case, DESA naturally separates synchronized binaries from young stars—two populations with nearly identical light curves—purely from their embedded positions in UMAP space, without requiring external kinematic or luminosity information. DESA thus offers a powerful new framework for multimodal, data-driven stellar population analysis, enabling both accurate prediction and novel discovery.

Unified Astronomy Thesaurus concepts: Stellar astronomy (1583); Neural networks (1933)

1. Introduction

Understanding the fundamental properties of stars is central to astrophysics, enabling insights into stellar evolution, galactic structure, and the conditions for planet formation. Traditionally, this effort was done using an analysis of different stellar measurements, such as stellar light curves (photometry) and stellar spectra (spectroscopy). While classical spectra analysis is usually used to predict stellar parameters related to absorption and emission lines such as $T_{\rm eff}$, $\log g$, $v \sin i$, and metallicity [Fe/H] (Y. Wu et al. 2014; A. E. García Pérez et al. 2016), classical light curves analysis usually uses spots modulation to find periodicity signatures (T. Reinhold et al. 2013, 2023; A. McQuillan et al. 2014; A. R. G. Santos et al. 2019, 2021; Y. Lu et al. 2020; S. Hattori et al. 2025) and magnetic activity (S. Mathur et al. 2014; Â. R. G. Santos et al. 2024). The revolution of deep learning models also affected stellar astrophysics, with various works using data-driven models that learn to predict or classify stellar parameters from observations and simulations. For example, K. Blancato et al. (2020), Z. R. Claytor et al. (2024), I. Kamai & H. B. Perets (2025a), and Z. R. Claytor & J. Tayar (2025) used deep learning models to predict rotational period from light curves. J.-S. Pan et al. (2024b) and X. Zuo et al. (2025) utilized transformer-based models to predict $\log g$. The use of machine learning (ML) in spectral analysis dates back to the pre-deep learning era, with the work of C. A. L. Bailer-Jones (2000). Since then, many works have been done combining deep learning and stellar spectra, with H. W. Leung & J. Bovy (2019), Y. Bai et al. (2020), R. Olney et al. (2020), H. W. Leung & J. Bovy (2023), X. Li & B. Lin (2023), and N. Koblischke & J. Bovy (2024) as some examples. For a review on ML in astronomy, please refer to G. Li et al. (2025).

Although the richness of works at the intersection between ML and stellar astrophysics, most of them use a single modality, and we refer to them as unimodal models. Multimodality models, on the contrary, try to combine the information from different modalities of the same object. This approach showed great success in natural language processing (NLP) and vision with works like CLIP (A. Radford et al. 2021) and its variants. Multimodality is of great importance in astrophysics—for example, photometry and spectroscopy reveal partial and very different stellar information; thus, accurate estimation of some stellar properties requires information from both modalities. Despite its importance, there are few works that utilize multimodal models in astrophysics. AstroCLIP (L. Parker et al. 2024) is one of such few attempts. The authors used galaxy images and spectra to train a CLIP-like model that aligns the representations of pretrained individual encoders. Maven (G. Zhang et al. 2024) is another example, in which simulated and real light curves and spectra of supernovae were used to create a multimodal model for supernova classification and redshift estimation. In addition, AstroM3 (M. Rizhko & J. S. Bloom 2025) is a model that uses light curves, spectra, and meta information for stellar variability classification.

Despite these early efforts, no existing model has yet demonstrated the full potential of multimodal integration for stellar astrophysics. In this work, we present the dual embedding for stellar astronomy (DESA) model—a large-scale, multimodal model that unifies spectroscopic and photometric data to produce robust, physically informative stellar embeddings. DESA introduces a novel alignment mechanism, DualFormer, which combines cross-attention, spectral duality constraints, and an eigenspace projection to produce a shared latent representation that preserves complementary information from both modalities. Importantly, we show that DESA's embeddings are not just useful for parameter inference but also encode rich astrophysical structure that enables meaningful clustering, similarity search, and physical interpretation.

We validate DESA on a wide range of astrophysical tasks, demonstrating both predictive performance and novel discovery capabilities. In zero- and few-shot experiments, DESA recovers color–magnitude and Hertzsprung–Russell (HR) diagrams with high accuracy, outperforming state-of-the-art unimodal and self-supervised baselines. Fine-tuning on challenging problems like binary detection and stellar age estimation yields substantial improvements over existing methods. Moreover, DESA's latent space naturally separates stellar populations such as synchronized binaries and young stars—two classes with nearly identical observables—without requiring external labels or kinematic measurements. These results illustrate that DESA is not only a high-performing model but also a general-purpose tool for discovering and interpreting stellar populations from large, heterogeneous surveys. A schematic description of DESA is depicted in the upper panel of Figure 1. In the sequel, we describe the DESA model, compare it with previous studies, and show its many advantages and successes.

This paper is organized as follows: In Section 2, we give general background on multimodal self-supervised learning (SSL); in Section 3 we present the details of the DESA model; in Section 4 we discuss the data set and preprocessing steps; in Section 5 we discuss implementation details; in Section 6 we shows the results of our model on various tasks and compare them to various baselines. Finally, Section 7 summarizes the findings and discusses future directions.

2. Multimodal SSL

Multimodal SSL is a subfield of SSL that focuses on leveraging multiple observations (modalities) of the same object, without relying on explicit labels. For example, this could involve aligning an image with its corresponding description, where the two modalities—vision and language—represent the same object. In the context of astrophysical measurements, multimodality typically refers to different surveys observing the same objects, such as a light curve and a spectrum (as discussed in this work) or a spectrum and image, as in L. Parker et al. (2024). The key advantage of multimodality over unimodality is the ability to uncover inter-modality relationships that may not be evident within each modality individually. These relationships can enhance performance on complex tasks that require information from both modalities. Given the surge in data volume and diversity across many scientific fields, coupled with a lack of corresponding labels, it is unsurprising that multimodal SSL has gained significant popularity in various domains (C. F. Brown et al. 2025; H. Cui et al. 2025). The primary challenge in multimodal learning lies in aligning the different modalities. There are numerous approaches to address this challenge—some specifically designed for multimodality, while others are more general self-supervised techniques. In the following, we discuss two of the most widely used self-

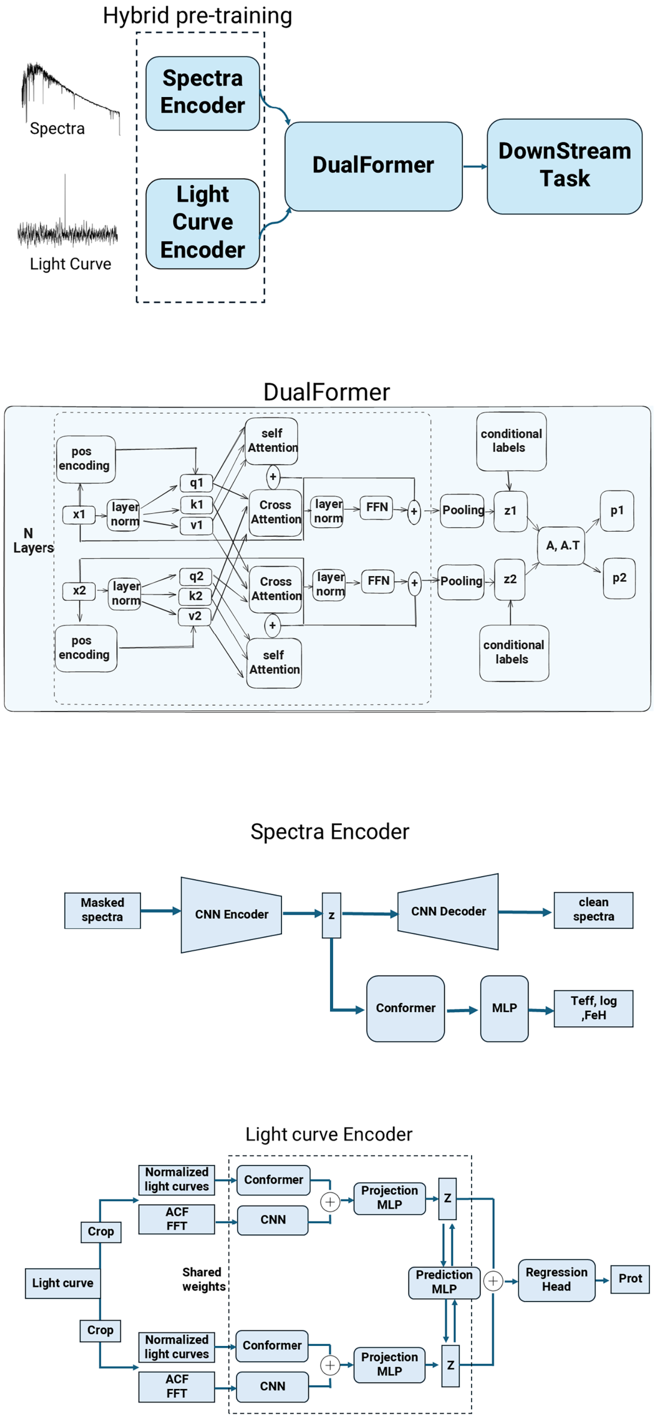

Figure 1. Upper panel—high-level diagram of the entire model. Lower panels—detailed diagrams of the DualFormer module, the spectra encoder, and the light-curve encoder.

supervised methods that can also be applied to alignment in multimodal scenarios.

2.1. Contrastive Alignment

Contrastive methods are a subset of SSL techniques that utilize the idea of "positive" and "negative" pairs, aiming to create an embedding space where the positive pairs are close to each other, while the negative pairs are pushed apart. This is performed by using special architectures and contrastive loss functions. Contrastive methods demonstrated great success in the computer vision domain with particularly influential works like SimCLR (T. Chen et al. 2020) and MoCo (K. He et al. 2019). While very popular, classical contrastive methods have drawbacks: they require a large batch size to sufficiently represent negative samples at each iteration. Also, the distinction between "positive" and "negative" pairs is a simplified assumption over the data. This is especially important in domains like astrophysics, where the transition between positive and negative pairs is often continuous. In addition, there might be more than one pair of the same object. In particular, in astrophysical data, having more than one spectrum for the same object is very common. Another common issue is the collapse phenomenon that occurs when the model ignores the inputs and creates identical and constant, trivially similar output vectors. Many works have tried to mitigate those drawbacks in different ways. X. Chen & K. He (2020) suggested a model called SimSiam with a modification to the classical infoNCE loss used in SimCLR. In SimSiam, the propagation of gradients is performed only through one side of the network. The authors showed experimentally that using this "stop-gradient" mechanism allowed the use of the positive pairs only while avoiding the collapse. MoCo uses a dynamical queue and a momentum-based encoder to reduce the need for a large batch size. Similarly, BYOL (J.-B. Grill et al. 2020) applies a slowly moving average update to one of the encoders. In the context of multimodality, CLIP (A. Radford et al. 2021) is a pioneering work that uses a contrastive approach to align text and images. As of today, contrastive methods are still among the most popular SSL frameworks for both unimodal and multimodal learning. As an example, AstroCLIP (L. Parker et al. 2024), Maven (G. Zhang et al. 2024), and AstroM3 (M. Rizhko & J. S. Bloom 2025) were trained using CLIP-like contrastive methods.

2.2. Regularized Alignment

A different line of work focuses on feature-level discrimination rather than instance-level discrimination, as in contrastive methods. This idea is motivated by canonical correlation analysis and was suggested as a self-supervised method by H. Zhang et al. (2021) and J. Zbontar et al. (2021). The latter was the motivation of the variance-invariance-covariance regularization (VicReg) architecture (A. Bardes et al. 2021). VicReg applies three different losses to prevent collapse and maintain alignment, ensuring the variance of the embeddings is sufficiently large (variance term), pushing the covariance between features to be the identity matrix (covariance term), and minimizing the L_2 distance between pairs of embeddings (invariance term). One advantage of this method is that it does not use negative pairs or asymmetric architectures. In their paper, the authors show that this model outperforms contrastive approaches on both unimodal and multimodal settings.

3. Multimodal Neural Network for Stellar Astrophysics

We propose a new framework for multimodal learning of stellar astronomy. Our model comprises two parts: individual encoders and an alignment module. In what follows, we present both components.

3.1. Hybrid Training of Individual Modalities

Similar to L. Parker et al. (2024) and G. Zhang et al. (2024), we start by training individual modalities separately. However, instead of only self-supervised training, we use a hybrid framework. The hybrid framework adds a supervised head to the self-supervised framework and trains the model with the following loss function:

$$\mathcal{L}_{\mathrm{hybrid}} = (1 - \lambda)\mathcal{L}_{\mathrm{ssl}} + \lambda\mathcal{L}_{\mathrm{sup}}, \tag{1}$$

where λ is a hyperparameter that balances supervision and self-supervision. The idea of using hybrid training comes from the fact that there are some stellar parameters that are known with good accuracy, and we can use those parameters to "guide" our model into more physical representations. This idea is not new and has already been used in M. Walmsley et al. (2022) for a unimodal galaxy model. In our case, the main role of supervision is to create good features at the end of each encoder, rather than the accuracy of the labels themselves. Nevertheless, we show that using this approach leads to state-of-the-art performance in nearly all tasks.

3.1.1. Spectra Encoder

To train the spectra encoder, we start with a convolutional neural network (CNN) encoder followed by two different channels, one for the self-supervised task and one for the supervised task. The self-supervised approach for spectra was chosen to be masked filling. As such, the input is a masked spectrum where 15% of the points are masked and replaced with either a predefined value (80% of the points) or a random value (20% of the points). The masked spectra are processed through the CNN encoder and then split into two branches: a CNN decoder that produces a filled spectrum, and a branch that consists of a conformer (A. Gulati et al. 2020) followed by a two-layer multilayer perceptron (MLP), which predicts T_{eff}, $\log g$, and [Fe/H]. A conformer is a transformer-based architecture with an added convolution between the multi-head self-attention layers. It was shown to be effective in capturing both global and local information, and was used by J.-S. Pan et al. (2024b) and I. Kamai & H. B. Perets (2025a) in their models. As suggested in J.-S. Pan et al. (2024b), we modified the standard conformer by using rotary position embedding (J. Su et al. 2021) instead of sinusoidal positional encoding. We chose $\mathcal{L}_{\mathrm{ssl}}$ to be the mean squared error (MSE) between the masked and filled spectra. $\mathcal{L}_{\mathrm{sup}}$ is a conformalized quantile regression (CQR) loss (Y. Romano et al. 2019). CQR is a form of quantile regression loss that mitigates one of the biggest problems of standard quantile regression—wrong prediction intervals. In CQR, quantile regression is combined with conformal prediction to create statistically calibrated confidence intervals. This is done by first calculating a "conformity score," which quantifies the error at each interval. This error is then used to calibrate the intervals on the test set.

One important aspect of stellar measurements is their signal-to-noise ratio (SNR): low-SNR samples are much harder to analyze. Ideally, we would like our model to be aware of those samples and attribute to them lower importance. To incorporate this information, we used the SNR of the spectra as a weight for the final loss during training. This ensures that low-SNR samples have less importance compared to samples with high SNR during training.

3.1.2. Light-curve Encoder

First, we preprocessed the light curves. Preprocessing is crucial because the raw Kepler measurements are noisy and uncalibrated. In addition, light curves themselves may not always be the best input for some tasks. For example, the period of a strictly periodic signal would be much more easily detected in the frequency domain than in the time domain (see, for example, the discussion in Z. R. Claytor et al. 2022). One complication with light-curve preprocessing is that different methods highlight specific information but usually hinder other types of information. For example, standard z-score normalization enhances periodicity but hinders activity-related patterns. Since we aim to create a general encoder, we decided to stack different normalizations and transformations into multiple input channels. We specify the details of light-curve preprocessing in Section 4.2. Since we trained the light-curve encoder in a contrastive-hybrid method, each light curve was augmented into two different views by cropping it at different times. For each view, we calculate the autocorrelation function (ACF) and the fast Fourier transform (FFT), which potentially encode important information (e.g., identification of periodic and quasiperiodic behaviors arising from stellar rotation of, e.g., eclipsing binaries) and add them as additional channels. Each view is then processed by a backbone that consists of two separate channels: a CNN encoder and a conformer module. The ACF and FFT channels are sent to the CNN encoder, and the normalized light curve is sent to the conformer encoder. The intuition behind this choice is that, in the frequency domain (ACF and FFT channels), periodicity is detected through peak detection. These tasks are “short range” and are typical for CNNs. Extracting meaningful information from the light curve requires some level of denoising and is more “long range.” Therefore, it is better suited for transformers (M. Morvan et al. 2022). This architectural choice is similar to the one presented in I. Kamai & H. B. Perets (2025a). They used the same conformer encoder and an LSTM instead of a CNN, and showed that using both the ACF and the normalized light curve together is better than using each one alone for period detection (see Table 2 in their paper). The embeddings are then combined and processed using a SimSiam method. The SimSiam framework can be described as follows: given the two outputs e_1 and e_2 of the backbone network, we apply an MLP network, f, such that $z_1 = f(e_1)$, $z_2 = f(e_2)$. We then project z_1 onto z_2 using another MLP g. The loss function is the cosine similarity between the projected and the unprojected features:

$$\mathcal{D}(z_1, z_2) = \frac{\langle g(z_1), z \rangle}{\|g(z_1)\| \, \|z_2\|}, \tag{2}$$

where $\langle \cdot,\cdot \rangle$ and $\|\cdot\|$ denote, respectively, the standard Euclidean inner product and the ℓ_2 norm. One important aspect of this loss is that the gradients flow only through z_1 and not through z_2. In the SimSiam paper (X. Chen & K. He 2020), the author refers to this as a “stop-gradient” mechanism. The final self-supervised loss is a symmetrized version of Equation (2):

$$\mathcal{L}_{\text{ssl}} = \frac{1}{2}\mathcal{D}(z_1, z_2) + \frac{1}{2}\mathcal{D}(z_2, z_1). \tag{3}$$

We add a two-layer MLP network on top of the final features to predict the rotational period, P_{rot}, when available, with $\mathcal{L}_{\text{sup}}$ as CQR.

3.2. DualFormer

The next step is to combine the embeddings from the pretrained individual encoders. Here we are using a novel approach specifically tailored for multimodality in stellar astrophysics. This is motivated by the observation that the information relationships between light curves and spectra of stars are very different from those found in language and vision modalities. While the text describes its corresponding image and vice versa, neither the light curve nor the spectra describes the other. They both partially describe the star, each from a very different perspective. Intuitively, light curves and spectra can be seen as orthogonal views of the star, as the former uses the time domain while the latter uses the frequency domain. Of course, the measurements are not mathematically orthogonal, since the measurements come from different surveys, with different bands and sensitivities, and can be taken at very different times, which complicates the relationship. Nevertheless, this intuition does not exist in other sets of modalities, such as image text or image spectra. Moreover, in both text and images (as well as spectra and images such as those in AstroCLIP), the dynamics of the system are not manifested in the data. In contrast, light curves measure time-dependent phenomena by design. This creates a time-dependent information relationship in the case of light-curve and spectra alignment.

Another uniqueness of astronomical data, not related to specific modalities, is the importance of prior knowledge. In astrophysics, one usually has some extra information about the observed objects. This can be, for example, stellar parameters that are known with good accuracy (inferred using classical analysis methods or previous unimodal ML models). As mentioned in Section 3.1, this information can be used to train individual encoders, but it can also be crucial during the alignment process, since this information is modality-invariant. These differences suggest that standard multimodal approaches might not be sufficient in our scenario, and that a specific model is needed.

The lower panel of Figure 1 presents a diagram of the alignment module, which we call DualFormer. The inputs are the final features from the light-curve and spectra encoders. They are first processed in a transformer-like module with a modified multi-head self-attention, but instead of self-attention, we use both self-attention and cross-attention, where the former focuses on intra-modality relationships, while the latter focuses on cross-modality relationships. This results in two feature branches with mixed information.

Next, we aggregate the information using average pooling, add conditional prior information, and project both features through the same linear layer denoted by the matrix A. This layer is the effective bottleneck of the network and should store the important shared information. Specifically, we use A to project the features, with one branch projected using A and the other branch using A^T. If the features were truly orthogonal, such a transformation would collineate them. To align the features while preventing collapse, we follow H. Zhang et al. (2021) and A. Bardes et al. (2021) with some important modifications. We adopt the same covariance loss that decorrelates the features

$$\mathcal{L}_{\text{cov}}(x, x') = \frac{1}{d}\sum_{k\neq l}[\text{Cov}(x, x')]^2_{kl}, \qquad (4)$$

where d denotes the vector dimension. However, unlike the original authors, we use the loss inside each branch and between branches; specifically, we decorrelate the features after projection, $p_1 = Az_1$ and $p_2 = A^T z_2$ in Figure 1. The full covariance loss assumes the form of

$$\mathcal{L}_{\text{cov}} = \mathcal{L}_{\text{cov}}(p_1, p_1) + \mathcal{L}_{\text{cov}}(p_2, p_2) + \mathcal{L}_{\text{cov}}(p_1, p_2). \qquad (5)$$

In addition, instead of a point-wise MSE loss between features, we use the following loss term:

$$\mathcal{L}_{\text{dual}} = \|\langle z_1, p_1\rangle - \langle z_2, p_2\rangle\|^2. \qquad (6)$$

$\mathcal{L}_{\text{dual}}$ can be seen as a less constrained version of the invariance term from A. Bardes et al. (2021): note that the loss is minimized when the quadratic forms $z_1^T A z_1$ and $z_2^T A^T z_2$ are equal. We see that while the standard invariance term requires z_1 and z_2 to be identical vectors, $\mathcal{L}_{\text{dual}}$ does not even require them to lie on the same hyper-surface (since in general $A \neq A^T$). This gives much more freedom for z_1 and z_2 to be different, but constrains the projections, namely, A and A^T. This is also why we chose to decorrelate the features after projection in $\mathcal{L}_{\text{cov}}$. Since A is not necessarily Hermitian, we expect the meaningful information to be stored in a shared vector space of A and A^T. Ideally, this would be the eigenspace of A. In Section 6, we show that this is indeed the case and that the eigenspace of A is an effective latent space that stores all the relevant information, as anticipated. Lastly, we do not use the variance loss term, as in A. Bardes et al. (2021), since $\mathcal{L}_{\text{dual}}$ requires fewer constraints on features, making the high-variance requirement redundant. The full training loss of the DualFormer is therefore

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_{\text{dual}} + \lambda\mathcal{L}_{\text{cov}}, \qquad (7)$$

where we set $\lambda = 0.5$.

Although this motivation may sound reasonable, it is not guaranteed to perform well in practice. We therefore test the architectural choices of DualFormer using an ablation study. Figure 2 shows two such studies. The upper panel shows a comparison of different attention mechanisms: self-attention (blue), cross-attention (red), and a combination of both (green). It can be seen that using both self- and cross-attention results in better performance in training and validation losses, as we might expect. The lower panel compares two cases: the first where, as described before, A and A^T are used to project the two features (red), and the second where A is used to project both features (blue). Again, we see that the chosen architecture (using A and A^T) outperforms the alternative. A general concern in multimodal models is that one modality will obscure the information from another modality. In our case, this might happen because the number of spectra samples used in the pretraining phase is ∼30 times larger than the number of light curves. As a consequence, the spectra encoder is larger than the light-curve encoder by roughly the same factor (see Section 5 for details), and we might have a situation where all the information in the final features comes effectively from the spectra features. First, we need to remember that for basic stellar properties, a spectrum is more informative than a light curve. Indeed, most basic stellar properties (e.g., T_{eff}, FeH, $\log g$, spectral type) are usually detected using spectral lines rather than light-curve features. Therefore, it makes sense that there would be more "spectra-related" information in the final features. However, we want to make sure that the light-curve information is not completely overwhelmed by the spectra information and that using the light curve improves the final performance, especially in tasks that require a sophisticated combination of different stellar parameters rather than basic stellar parameters. In Sections 6.3 and 6.4, we test such tasks and see that using the combined model significantly improved the performance compared to spectra-only or light-curve-only encoders. Another possible test is a sensitivity analysis, not related to a specific task, that identifies which features are most influenced by each modality. For that, we performed the following experiment: We calculated the final features of the test set 3 times. One time with the full input (light curve and spectra), one time with the light curve replaced with zeros (we call this case spectra only), and one time with the spectra replaced with zeros (light curve only). We then look at the difference between the full features and spectra-only/light-curve-only features. The difference is calculated per feature and averaged across all samples. This way, we can trace each individual feature and see for which modality (light curve or spectra only) the difference is smaller. This modality is the more dominant for this specific feature. Figure 16 in the Appendix shows the results of such an experiment. It shows the per-feature difference for both light-

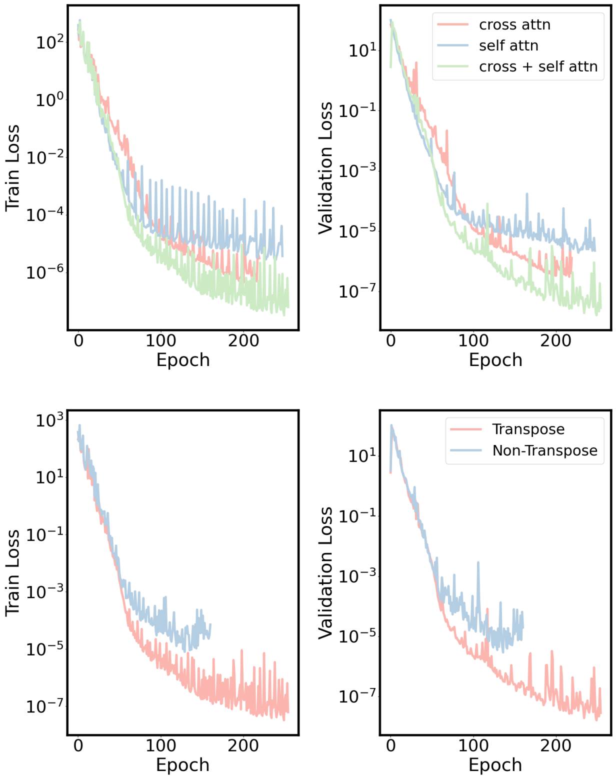

Figure 2. Ablation study results. Upper panel: different attention mechanisms. Lower panel: different uses of A—with and without A^T for one of the projections (refer to Section 3.2 for details).

curve-only (gray) and spectra-only (red) features. The open blue circles mark the indices where the light-curve-only features are more dominant. There are 75 circles, which correspond to ∼30% of the features, an order of magnitude more than the ratio of light-curve to spectra samples and number of parameters. This is not negligible and supports the results we find in Sections 6.3 and 6.4. Interestingly, we see that while there are small areas with clear dominance of one modality (around indices 150, for example), in most cases both modalities seem important. This is especially true for the last indices that correspond to the added prior information. We conclude that while there is more spectral information in the final features, the contribution of the light-curve encoder is not negligible and especially important in complicated tasks that require combining different information types. We note that while this experiment is convincing, it is not as rigorous as a full ablation study where we completely remove different components and retrain the model. Such an experiment could be explored in future work.

4. Data

We train the full model using low-resolution spectra from LAMOST (G. Zhao et al. 2012; C. Wang et al. 2022) DOI: 10.12149/100632, and light curves from Kepler S. Mathur et al. (2017) DOI:10.17909/T9488N.

4.1. Spectra Preprocessing

LAMOST is a low-resolution ($\frac{\lambda}{\Delta\lambda} \sim 1800$) spectrometer with two arms: a blue arm that covers a wavelength range of 3700–5900 Å, and a red arm that covers a wavelength range of 5700–9000 Å. The LAMOST survey consists of millions of spectra of stars, galaxies, and quasars. We used LAMOST DR8 and downloaded all the spectra of A, F, G, K, M stars with $3000 \leqslant T_{\rm eff} \leqslant 7500$ K. We removed samples without measured $T_{\rm eff}$, $\log g$, or [Fe/H] (from the LASP pipeline; Y. Wu et al. 2014), and samples with nonsensical negative SNR. This resulted in about 6.5 million samples. As mentioned in Section 3.1, the hybrid approach does not require having supervised labels. Here, we decided to use only samples with labels, given the large size of the data set available. The preprocessing of spectra is similar to what was suggested in StarGRUNet (X. Li & B. Lin 2023): we first translate the wavelength to the rest frame using radial velocity measurements from the LASP pipeline (Y. Wu et al. 2014). Next, we divide the spectrum into blue and red regions and resample each region using linear interpolation. We then apply a median filter to each region, followed by a continuum normalization using a fifth-order polynomial, and a final step where we remove points higher or lower than $\pm 3\sigma$ and normalize the flux to have zero mean and unit standard deviation. For a more detailed explanation of the different preprocessing steps, please refer to X. Li & B. Lin (2023). Figure 3 shows an example of the preprocessing stages of LAMOST spectra.

4.2. Light-curve Preprocessing

Kepler was a space mission designed to provide high-cadence light curves for stars and for the search for exoplanets through transits. As such, Kepler measured the light curves of around 200,000 main-sequence and giant stars for almost 4 yr with a cadence of approximately 30 minutes. We used Kepler samples with stellar properties from T. A. Berger et al. (2020), resulting in 183,435 samples. The preprocessing of the light curves follows two different ways: one using the absolute magnitude and the other using the mean and standard deviation. As mentioned in Section 3.1.2, the reason for the two normalizations is that different properties require different light-curve information. The activity and luminosity, for example, are related to the absolute amplitude of the light curve (normalized by the distance). On the other hand, the period is more easily detected using standard normalization, like zero mean and unit standard deviation. The absolute magnitude normalization was done by first dividing the light curve by 2^{-k} where k is the absolute magnitude, calculated using the Kepler magnitude, KMAG, and the distance from T. A. Berger et al. (2020). In addition to the raw light curve, as mentioned earlier, we also calculated the FFT and the ACF, which help to determine the period. Figure 4 shows an example of the preprocessing stages of the Kepler light curve.

5. Implementation Details

In big models, like DESA, hyperparameter tuning can be a very challenging task. This becomes even harder when the model consists of two steps—pretraining and alignment. We therefore chose to use a simple heuristic when defining the hyperparameters of our model. As the number of spectra samples is much larger than the number of light-curve samples (6.5 million versus 200,000), we designed the individual encoders such that the spectra encoder has more parameters than the light-curve encoder. This is motivated by neural scaling laws, a phenomenological relationship between data set size, model parameters, and performance, which was originally found in the vision and language domains (J. Kaplan et al. 2020; J. Hoffmann et al. 2022), but recent works have shown that it also applies to astronomical data (J.-S. Pan et al. 2024a; M. Walmsley et al. 2024). Therefore, the dimension of the final spectra features was chosen to be 2048, and that of the light curve was chosen to be 256. The number of parameters in the spectra and light-curve encoders is about 500 million and 11 million, respectively. During hybrid training, λ was chosen arbitrarily to be 0.5. The embedding dimension in DualFormer was set to 256, and the number of parameters in this module is also about 11 million. The light-curve encoder was trained using a learning rate decay scheduler with the cosine annealing method. The initial learning rate is $2 \cdot 10^{-5}$, decreasing to $2 \cdot 10^{-6}$. All other modules were trained with a constant learning rate of $2 \cdot 10^{-5}$. We trained all modules with the AdamW optimizer (I. Loshchilov & F. Hutter 2017). Lastly, we estimated the energy used to train the entire model using the CodeCarbon[5] package. It is estimated to be 334 kWh for the entire model, out of which 204 kWh are for the pretraining stage. All of the code used for training and experiments is publicly available on GitHub[6] and on Zenodo (I. Kamai et al. 2025).

6. Results

6.1. Pretraining Results

First, we present the results of the hybrid pretraining of individual modalities. As mentioned before, the spectra encoder was trained using $T_{\rm eff}$, $\log g$, and [Fe/H] labels from

[5] https://codecarbon.io/

[6] https://github.com/IlayMalinyak/DESA

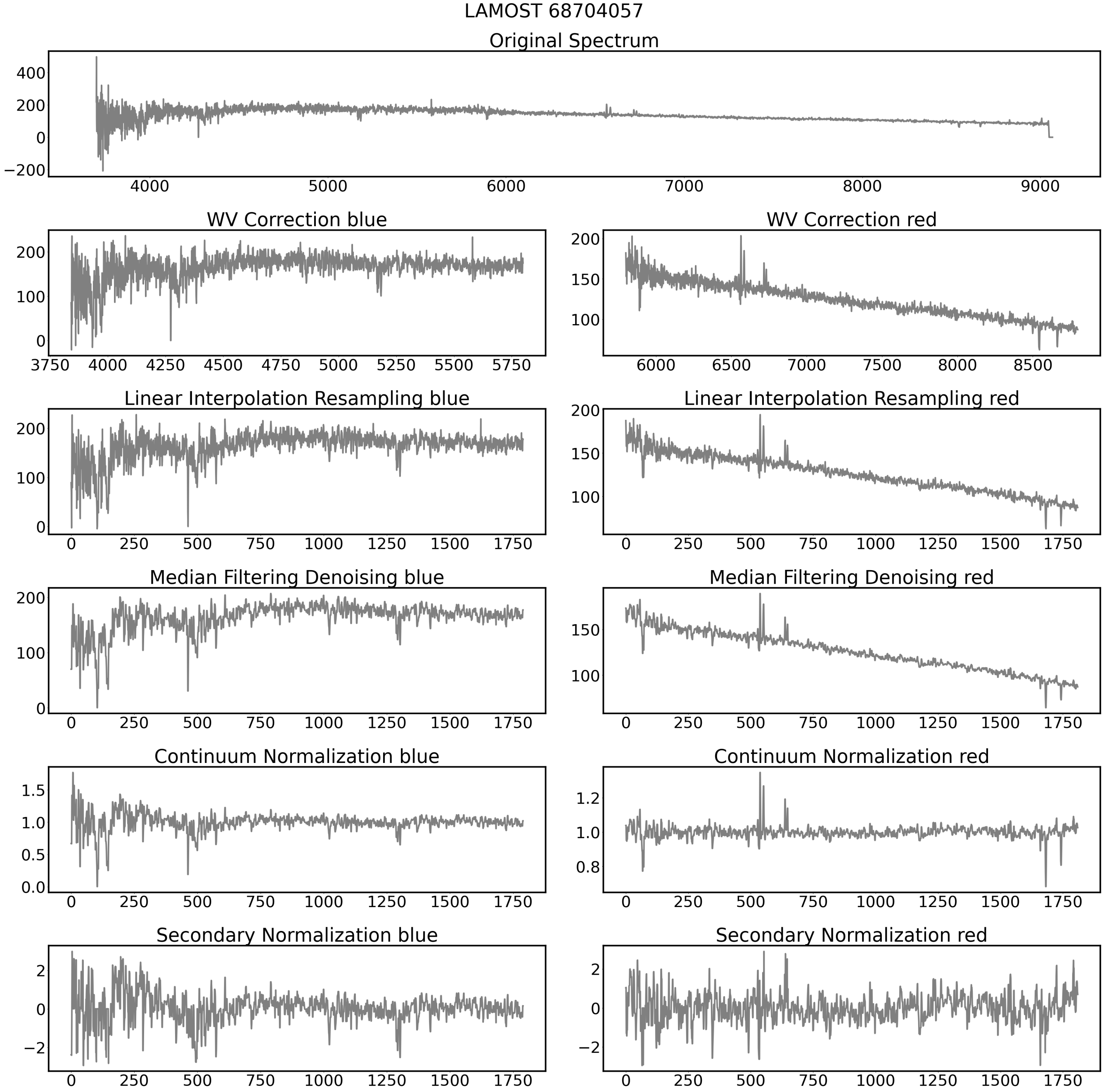

Figure 3. Example of the preprocessing steps for LAMOST spectra. The left column is the blue range, and the right column is the red range.

the LASP pipeline. However, we test the results using labels from the high-resolution APOGEE survey (Abdurro’uf et al. 2022), at the same temperature range. This is a common practice and was done in previous works such as StarGRUNet (X. Li & B. Lin 2023). The results on the test set of both LASP labels and APOGEE labels are shown in Figure 5. For each label, we also plot the 80% and 50% confidence intervals. It can be seen that the mean average error (MAE) on LASP is a factor of ∼2 lower compared to APOGEE for all labels. This is reasonable given the fact that the APOGEE labels come from high-resolution spectra, while the spectra that are given to the model are low resolution. Another interesting difference between the labels is related to the prediction intervals. As mentioned in Section 3.1, the prediction intervals were calibrated on a held-out validation set, different from the test set. This validation set uses the LASP labels. We see that for the LASP test set, this calibration results in the desired coverage (80% of the points are inside the 80% interval, for example), while for the APOGEE labels, the coverage is lower. This is another evidence that there is a significant difference between the labels. We also compare our results with StarGRUNet. StarGRUNet (X. Li & B. Lin 2023) used a GRU-based model to predict stellar parameters on a wide range of SNR values ($\mathrm{SNR} > 5$). While there are works with better performance for a specific range of SNR (X. Li et al. (2022), for example, who used $10 < \mathrm{SNR} < 20$), this is the

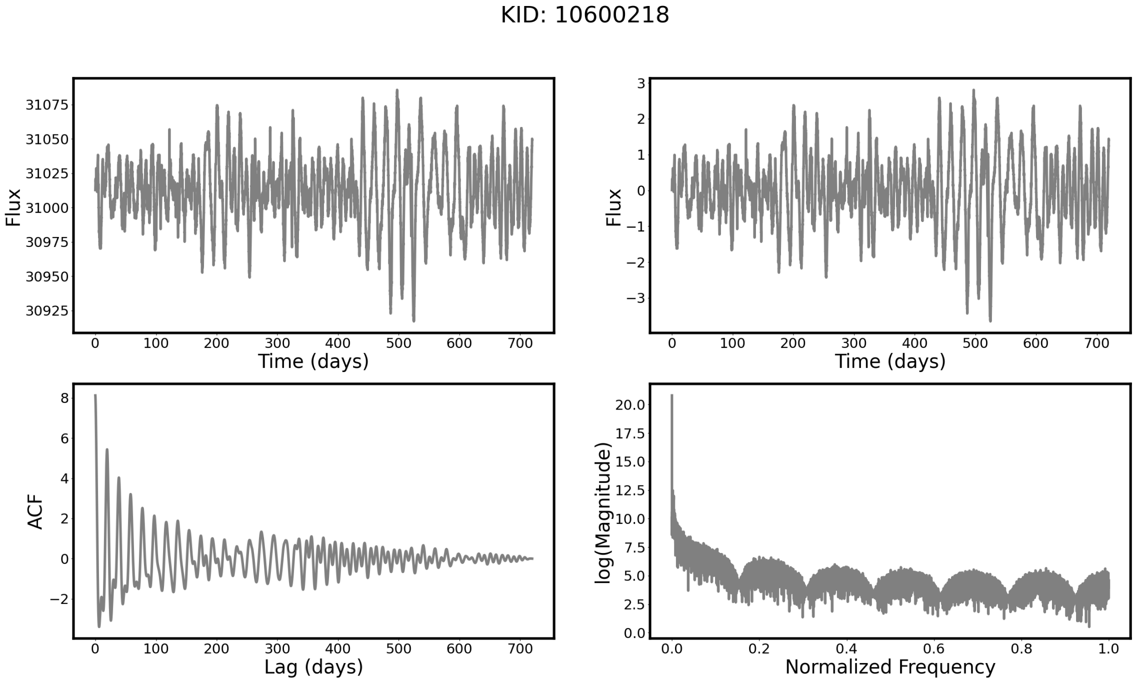

Figure 4. Example of the preprocessing steps for Kepler Light curve. The upper row shows the raw light curve normalized by absolute magnitude (left) and mean and standard deviation (right). The lower row shows the ACF (left) and FFT (right).

state of the art for a model that uses LAMOST low-resolution spectra and a wide range of SNR values.

Table 1 shows a comparison between StarGRUNet and DESA. We note that the results are very close. Specifically, our model performs slightly better than StarGRUNet for $T_{\rm eff}$ and [Fe/H], and slightly worse for $\log g$. However, it is worth mentioning that we used SNR > 0, and StarGRUNet used SNR > 5. This seemed to be a result of the fact that we incorporated the SNR information into the training. In Figure 14 in the Appendix, we show MAE versus different SNR bins for $T_{\rm eff}$, $\log g$, and [Fe/H]. We see that the sensitivity of our results to SNR is indeed better than StarGRUNet (see Figure 11 in their paper). However, there is still nonnegligible sensitivity for both LASP and APOGEE labels. While it is probably impossible to remove all the dependency on the noise, this might be further investigated and improved in future papers.

The light-curve encoder was trained using periods aggregated from all recent catalogs (A. McQuillan et al. 2014; A. R. G. Santos et al. 2019, 2021; T. Reinhold et al. 2023; I. Kamai & H. B. Perets 2025a). This results in 104,433 samples with a period label. The results are shown in Figure 6. We can compare the results to the work of K. Blancato et al. (2020), which trained a CNN network on data from A. McQuillan et al. (2014). In their work, they reported a root mean squared error (RMSE) of 5.2 days. Our RMSE is 2.61 days, a factor of 2 lower.

6.2. Zero- and Few-shot Results

Next, we test the full DESA model. We train the full model using the pretrained individual encoders. The conditional labels that were added during training are the labels used for pretraining ($T_{\rm eff}$, $\log g$, [Fe/H], $P_{\rm rot}$), with the addition of the radius, R, and renormalized unit weighted error (RUWE), which measure the error of the astrometric fit from Gaia. The last two were taken from T. A. Berger et al. (2020). Similar to the hybrid method, we do not require those labels to exist.

Before discussing the fine-tuning results, we investigate the learned feature space. As mentioned in Section 3.2, the linear layer A is an effective bottleneck of the network, and we want to utilize the information stored in it in a way that is invariant to the transpose operation. Therefore, our final feature space is the projection of the feature vectors (z_1, z_2 in Figure 1) onto the eigenspace of A. This can be written as

$$f = (z_1 + z_2)^T V, \tag{8}$$

where z_1, z_2 are the preprojection feature vectors and V are the eigenvectors of A after training.

We compare our model with contrastive and regularized self-supervised methods that achieved state-of-the-art results in various tasks: VicReg, SimSiam, and MoCo. Each of the methods represents a different methodology—VicReg is a regularized method, SimSiam is a "positive-only" contrastive method, and MoCo is a "positive and negative" contrastive method. The use of positive and negative pairs in our scenario might be challenging because there are many samples with multiple spectra. This means that in a batch of samples, we might have off-diagonal positive pairs, which means that they would count as negative pairs. To overcome this, we created a version of MoCo with a special sampler that ensures the uniqueness of stars in each batch. We call this variant MoCo-clean. To make sure that all models get the same information, we added the same conditional labels to all models.

First, we want to compare the final features of the different models. This is done by a UMAP (L. McInnes et al. 2018) dimensionality reduction of the final features. Figure 7 shows the UMAP of our model and all other alternatives, calculated only on the test set. The UMAP is colored in two ways. The upper panel shows coloring that corresponds to the stellar luminosity from T. A. Berger et al. (2020). It can be seen that

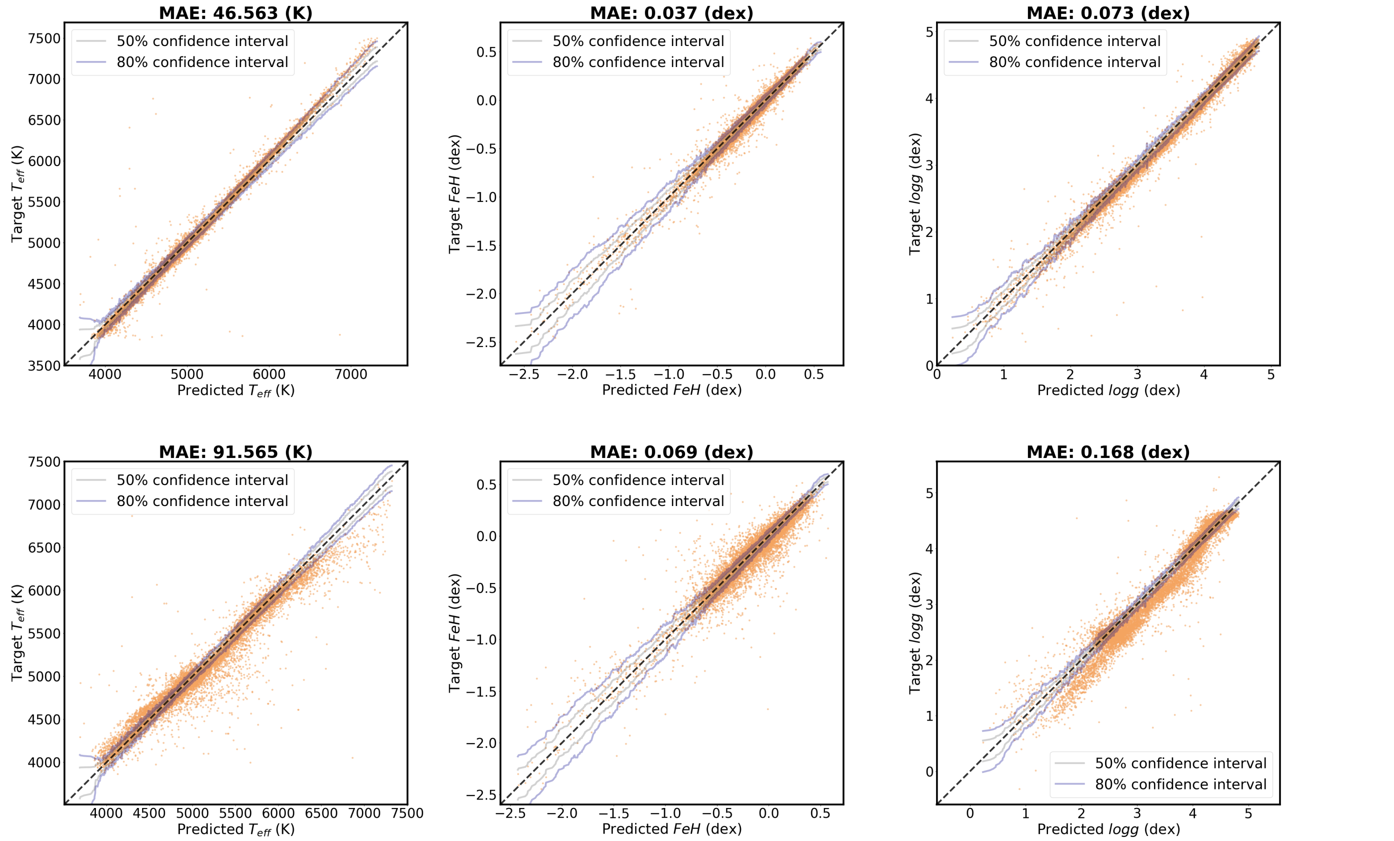

Figure 5. Upper panel—results of the spectra encoder on LASP labels. Lower panel—results on the labels from APOGEE. The purple and gray lines represent prediction intervals of 80% and 50%.

Table 1
Result of Spectra Encoder

Model	T_{eff} MAE (K)	log g MAE (dex)	[Fe/H] MAE (dex)
DESA (ours)	**91.56**	0.168	**0.069**
StarGRUNet	93.77	**0.162**	0.070

Note. Ground truth labels are from APOGEE. See text for details. The columns correspond to the Mean Average Error (MAE) in effective temperature, surface gravity, and metallicity. Bold values represent the best result.

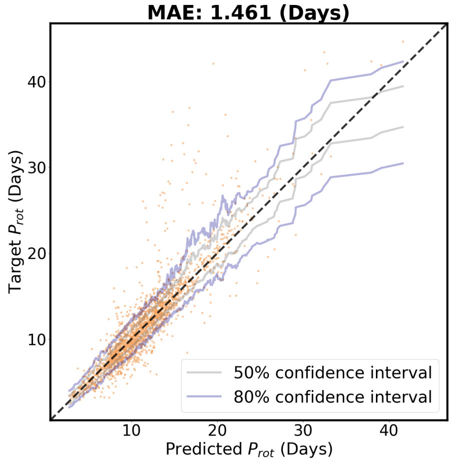

Figure 6. Results of the light-curve encoder. Gray and purple lines represent 50% and 80% intervals.

our model shows a smooth change in color, which reflects a strong correlation between the latent features and luminosity. Looking at other models, we see that vanilla MoCo shows a thin manifold in UMAP space, which implies a collapse mode. MoCo-clean and SimSiam do not show a strong correlation with luminosity, and VicReg is the only model that shows a smooth correlation. The lower panel shows colors that correspond to stellar classes derived by D. Godoy-Rivera et al. (2025), based on a position on a color–magnitude diagram (CMD). The classes are dwarfs, giants, subgiants, overlap dwarf/subgiants, photometric binaries, and "uncertain MS," which correspond to samples that sit below a lower envelope on the CMD. It can be seen that our model creates the best natural clustering with different stellar types lying in different areas of the UMAP space. This is significantly better than all other models, which mix different types, usually dwarfs and subgiants. Next, we would like to test the consistency of the final embeddings. This is done using stars with multiple spectra measurements. Those stars have different instances in the test set that reflect the same stellar object. We therefore expect them to have very similar embeddings and to sit close to each other in UMAP space. First, we found all stars with a pair of samples in the test set. Then, for each star, we calculated the UMAP distance between the stars in the pair. Figure 8 shows histograms of distances of pairs for all models. It can be seen that the distances of our model are significantly lower compared to all other models. Specifically, the average

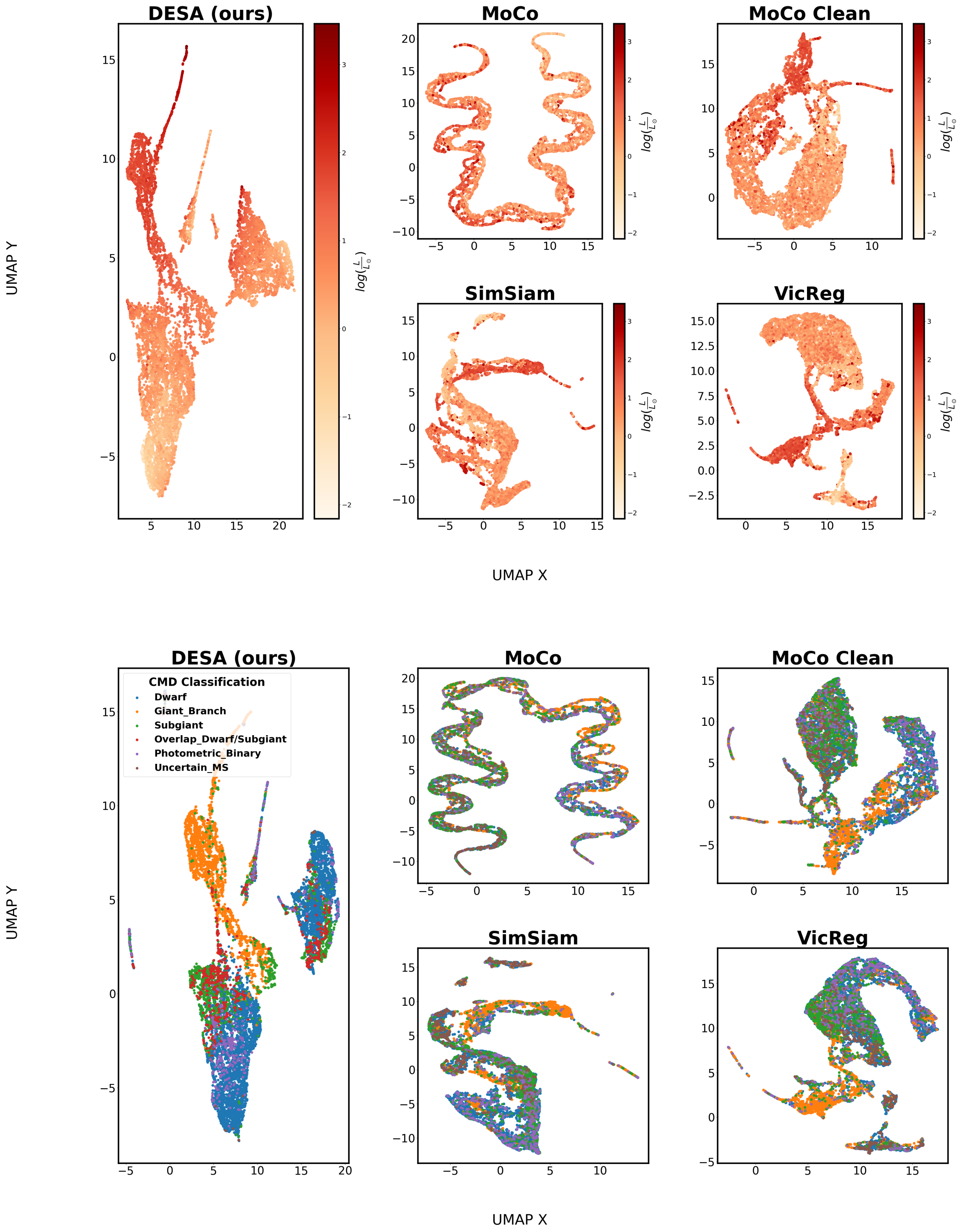

Figure 7. UMAP of the final features of different models. In the upper panel, color represents luminosity. In the lower panel, the color represents CMD classes from D. Godoy-Rivera et al. (2025).

distance of our model is more than 2 orders of magnitude lower than all alternatives. This can also be shown visually by looking at the UMAPs and highlighting pairs. Such plots are shown in Figure 15 in the Appendix, showing 10 example pairs. We conclude that the latent features of our model are significantly more meaningful and consistent compared to all other models.

We now move to a more quantitative comparison, using zero-shot and few-shot experiments. The zero-shot experiment consists of applying a simple clustering algorithm on the

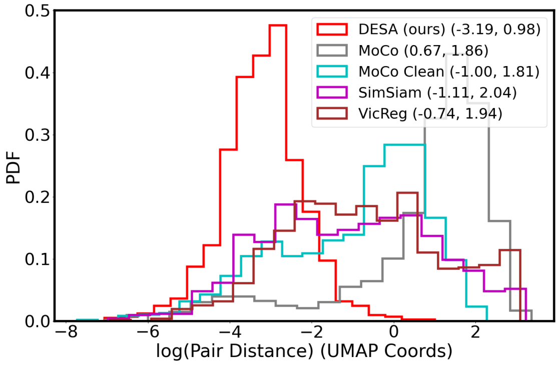

Figure 8. Histograms of UMAP distance of stars with multiple spectra. The legend shows the model name and the mean and standard deviation of the distribution in parentheses.

Table 2
Result of Zero-shot CMD Clustering (First Column) and Few-shot Regression

Model	Zero-shot accuracy	Linear Regression R^2	BP − RP Accuracy	*G*mag Accuracy
DESA (ours)	**0.40**	**0.920**	**0.677**	**0.711**
MoCo	0.18	−0.001	0.208	0.159
MoCo-clean	0.11	−0.0004	0.202	0.160
SimSiam	0.15	−0.0003	0.202	0.158
VICReg	0.25	−0.0003	0.202	0.158

Note. See the text for details. The Zero-shot column represents the accuracy of UMAP clustering. The Few-shot columns show the accuracy of color and magnitude predictions and R^2 of the combined prediciton. Bold values represent the best result.

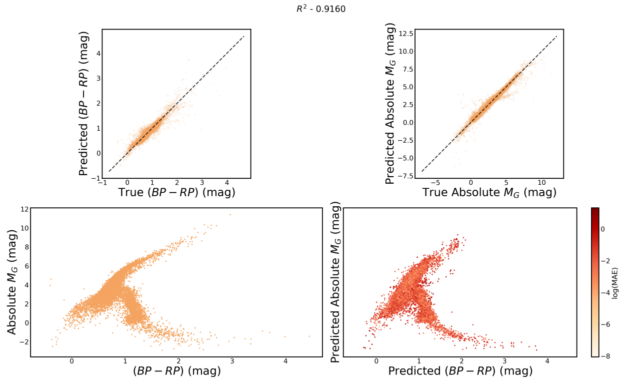

Figure 9. Result of few-shot learning of a linear regression model. The two upper panels show the predicted labels vs. the true labels. The lower left panel shows the true CMD. The lower right panel shows the predicted CMD. Colors on the lower right panel correspond to the MAE.

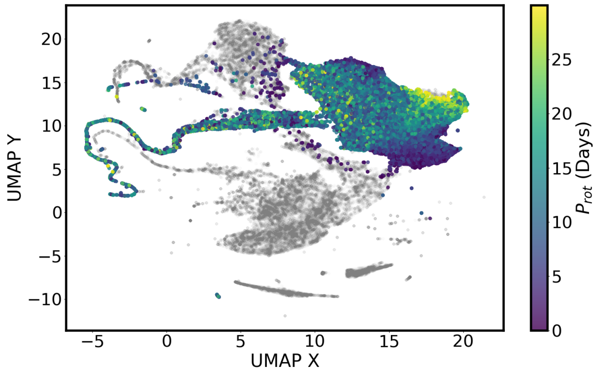

Figure 10. UMAP of final features of the DESA model. All points are in gray. Colored points correspond to samples with $P_{\rm rot} < 30$ days in one of the latest period catalogs (A. McQuillan et al. 2014; A. R. G. Santos et al. 2019, 2021; T. Reinhold et al. 2023; I. Kamai & H. B. Perets 2025a).

UMAP-reduced features, with classes corresponding to the CMD classes from D. Godoy-Rivera et al. (2025). The clustering algorithm is a Gaussian mixture model (GMM). The few-shot experiment consists of applying a simple linear regression model on 20% of the test set (∼2500 samples) to predict color (dereddened BP − RP) and magnitude (absolute dereddened *G*-band magnitude), both from D. Godoy-Rivera et al. (2025). We measure the accuracy of the zero-shot classifier as well as the R^2 and accuracy (defined as the number of points within 10% absolute error) of the few-shot regressor. The results are summarized in Table 2. It can be seen that our model outperforms all other models with a very large margin on all metrics. Specifically, the R^2 of all alternative models is around zero. This means that there is no linear relationship between their features and the desired labels. Our model shows a strong relationship with $R^2 = 0.92$. This point is further demonstrated in Figure 9, where we plot the prediction results. It can be seen that the model learns not only each parameter alone, but also the correct relationships between them, which create an effective CMD. The last point suggests that our final features can be easily fine-tuned, using a few-shot learning, to reproduce known stellar diagrams and can serve as a learned general diagram. We call these diagrams *neural diagrams*. Another example of few-shot learning and recovery of the HR diagram can be seen in Figure 17 in the Appendix. To conclude this subsection, we provide an example of the type of insights that result from explorations of *f*, the final feature space of DESA. For that purpose, we calculate the UMAP of *f*, calculated using Equation (8), on the entire data set. Figure 10 shows the UMAP of the entire data set, with samples that have $P_{\rm rot} < 30$ colored by their period. It is easy to notice a unique structure in the position of samples on the UMAP space, considering their period. There are two main blobs of short periods, located on either side of the main shape. This is interesting given the fact that short-period stars are usually a mixed population of very young stars, which are fast because they are not yet affected by magnetic braking, and synchronized binaries, which are fast because of tidal synchronization. Separating the populations is a hard task and involves different methods. One way is to look at the luminosity excess of stars—we expect binaries to be overluminous compared to single stars with the same parameters. Another way is by looking at the kinematics of stars—since the peculiar velocities of stars are excited by encounters with spiral arms and molecular clouds, the velocity dispersion of young stars should be lower than that of old stars (i.e., the velocity dispersion provides a "kinematic age"). In our previous paper, I. Kamai & H. B. Perets (2025b), we analyzed a population of fast-rotating stars and separated them into groups of binaries and young stars using these two approaches. To see if our model creates such separation naturally, we use their data of fast rotators and plot it in UMAP space. Figure 11 shows such plots with different coloring. In the upper left panel, we color the fast

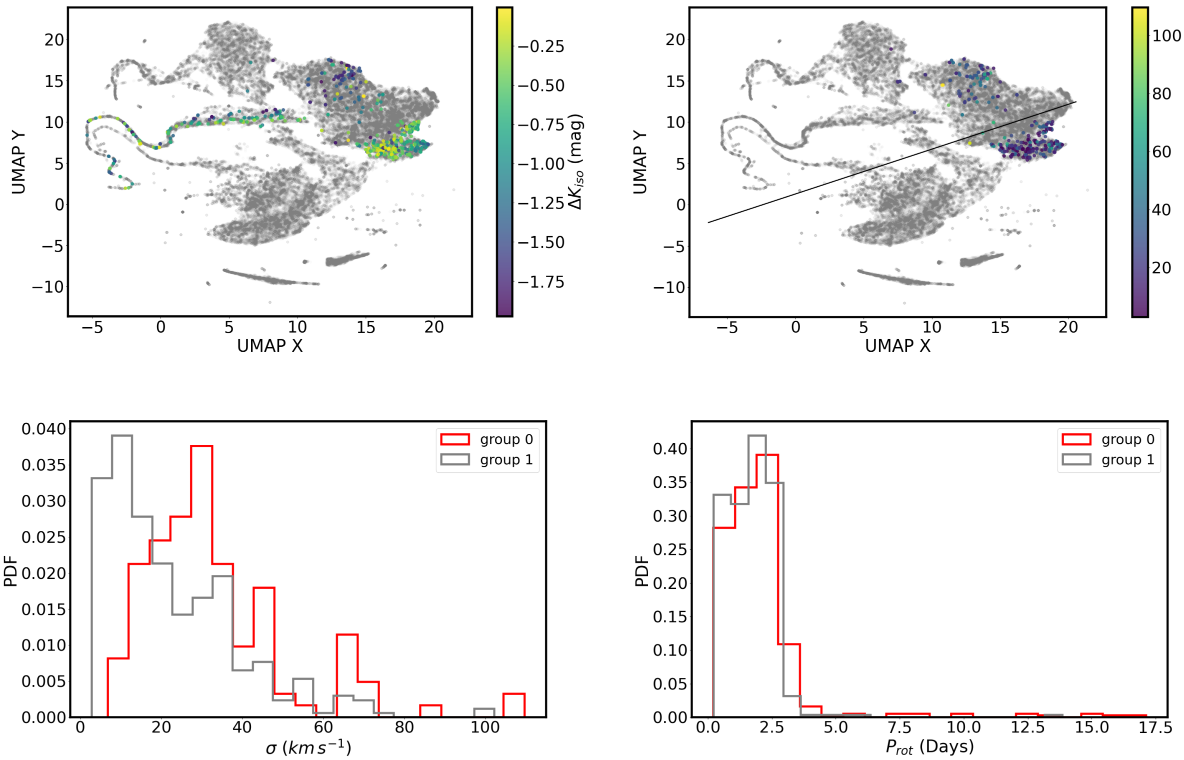

Figure 11. UMAP of fast rotators from I. Kamai & H. B. Perets (2025b). Upper left—colors correspond to the luminosity excess from a single-star model ($\Delta K_{\rm iso}$, see I. Kamai & H. B. Perets (2025b) for details). Upper right—color corresponds to peculiar velocity dispersion (σ) from D.-C. Chen et al. (2021). Note that not all samples have a σ value. The black line is a separation line found by a linear SVM classifier and a clustering of the points into two classes. Lower left—σ histograms of the two clusters from the upper right panel. Lower right—$P_{\rm rot}$ histograms of the same clusters.

rotators according to their luminosity excess, compared to a single star with the same parameters. This difference is called $\Delta K_{\rm iso}$. The lower the value of $\Delta K_{\rm iso}$, the more overluminous the star. For more details on how we calculated $\Delta K_{\rm iso}$, please refer to I. Kamai & H. B. Perets (2025b). We can see that the points are separated into three distinct areas—the two blobs on either side of the main structure, which we refer to as the "mainland," and the long and thin "tail" that goes out of the main structure. We note that this long tail is an area of stars with large RUWE (UMAP with RUWE colors is shown in Figure 18 in the Appendix), and therefore possibly binaries. In the mainland, the two populations are clearly distinguished by their $\Delta K_{\rm iso}$. This is further demonstrated in the upper right panel, which shows their kinematics. Namely, their peculiar velocity dispersion, σ, taken from D.-C. Chen et al. (2021). It can be seen that only samples on the mainland have a σ value, with essentially the same separation as in the upper left panel, suggesting that samples on the upper blob are synchronized binaries and samples on the lower blob are young. To further test this assumption, we separate the two blobs using a GMM clustering and a linear support vector machine (SVM) classifier. The separation line can be seen as the black line in the upper right panel. The lower panels show the distributions of σ (left) and $P_{\rm rot}$ (right) for the two clusters. We can see that while the two groups have almost identical rotation period distributions, their kinematic distributions are very different, justifying the fact that these are indeed separate populations of synchronized binaries (upper blob with high σ and low $\Delta K_{\rm iso}$) and young stars (lower panel with low σ and high $\Delta K_{\rm iso}$). The fact that these populations are naturally separated by the model has far-reaching implications. For example, it means that we do not need $\Delta K_{\rm iso}$ and σ measurements, which are prone to different errors, to separate the groups. All we need is a light curve, a spectrum, and a period measurement. Then, we can conclude with high confidence if a star is young, only by looking at its position on the UMAP space. This demonstrates that the embeddings of DESA can be a very powerful and very flexible tool for different stellar population analysis tasks. Next, we test DESA on challenging fine-tuning tasks.

6.3. Binary Detection

Here, we test our model in a full fine-tuning scenario. We add a small transformer prediction head and fine-tune the model on a more challenging task, binary detection. Roughly half of the stars are part of a binary or higher-multipole system (D. Raghavan et al. 2010). However, detecting them might be very challenging. There are many methods to detect binaries, with different sensitivities. If, for example, the orbital plane of the binary is perpendicular to the line of sight, we would call them "eclipsing binaries" since we would see the stars eclipse each other. Then it would be easier to detect them using light-curve measurement. Another example is the use of spectroscopy to measure the changes in radial velocity that come from the gravitational pull of the companion. This is stronger when the stars are in close orbit. Since every method is sensitive to only a subset of binaries, it is crucial to combine photometric and spectroscopic information to accurately predict a wide range of stars. To create a data set, we again use the data in D. Godoy-Rivera et al. (2025). They reported more than 30,000 binaries in the Kepler field, which resulted from different detection methods. Specifically, to create a data set

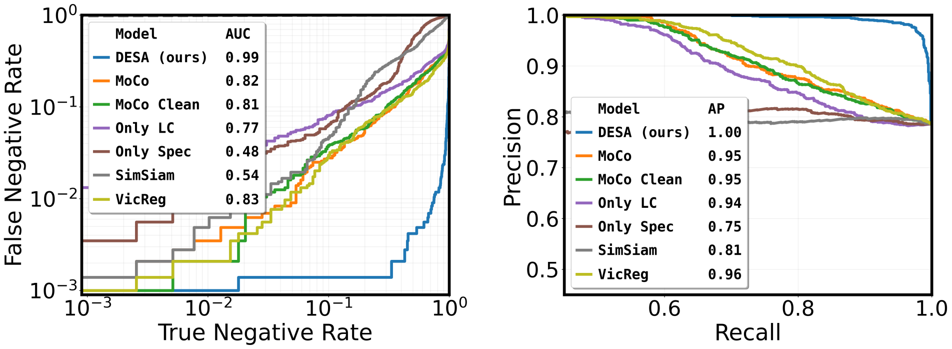

Figure 12. Experimental results of binary detection for different models. The left panel shows TNR vs. FNR. The right panel shows precision–recall curves.

for binary detection, we need a high-confidence sample of binaries and a high-confidence sample of singles. While the first one might be trivial, the second is more challenging, since there are probably many undetected binaries in Kepler. Therefore, to create a sample of single stars, we selected stars satisfying the following four criteria:

1. Not flagged as binaries by D. Godoy-Rivera et al. (2025) (flag binary union).
2. Have RUWE < 1.2.
3. Not flagged as potential synchronized binaries by I. Kamai & H. B. Perets (2025b).
4. Have $|\Delta K_{\mathrm{iso}}| < 0.3$.

The last point is motivated by the fact that binaries are expected to be more luminous compared to single stars with the same temperature, and ΔK_{iso} refers to the difference between the expected magnitude of a single star and the measured absolute magnitude, as mentioned in Section 6.2. To create a sample of binaries, we took samples that were flagged by D. Godoy-Rivera et al. (2025) as binaries using the most common methods: RUWE, Gaia Non-single-star catalog (NSS), RV variable, and eclipsing. This results in a data set of about 14,000 binaries and 4000 singles.

We fine-tuned our model, with an added classification head, for a binary classification of binarity—all the binaries were assigned as class 1, regardless of the detection method, and all the singles were assigned as class 0. As baselines, we trained the same models that were used in Section 6.2 as well as unimodal models. The unimodal models are simply the pretrained encoders with an added prediction head. Using the pretrained encoders is important as a sanity check—we want to see if our model is able to extract meaningful information from the combination of modalities and, specifically, better than each modality alone. In all models, we made sure that the number of trainable parameters is not less than that in our DESA model (11 million). Figure 12 compares the results using a precision–recall curve (right panel) and a true negative rate (TNR) versus a false negative rate (FNR) curve (left panel). Notice that a TNR–FNR curve can be seen as a flipped Receiver operating characteristic (ROC) curve, and therefore, the area under the curve (AUC) metric has the same meaning (with the area calculated in the opposite direction). Note that we opted for the TNR–FNR curve instead of the standard ROC curve for visual clarity. It can be seen that our model performs significantly better compared to all other models, with an AUC of 0.99 and average precision (AP) of 1.00. Figure 13 shows the confusion matrix of our model. It is interesting to compare it with the confusion matrices of all other models. Such a comparison revealed that, except for VicReg, all other models collapsed to predict the majority class (binary class). With the exception of the proposed DESA, VicReg was the only model able to predict some single stars correctly, but it did so with much lower precision for that class, in comparison to our model. The precision of VicReg on the single-star class is 48%, while our model achieved a precision of 88% on that class. Confusion matrices of all other models are reported in Figure 19.

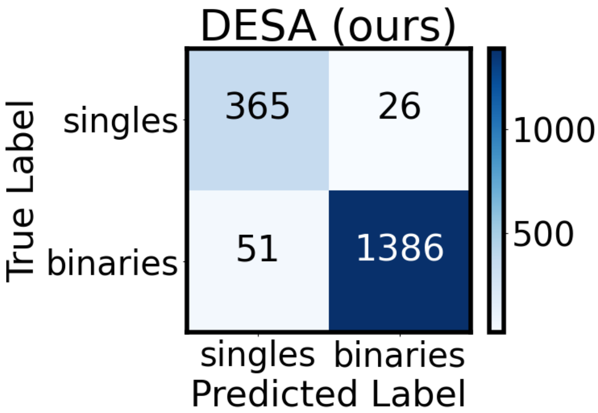

Figure 13. Confusion matrix of our model on the binary detection task.

6.4. Stellar Age Prediction

Our last experiment is fine-tuning for stellar age prediction. Age is one of the most challenging stellar properties to detect. Since age is not directly measurable, people use relationships with other properties to infer ages. However, these relationships might not be trivial. One such example is gyrochronology, which started as the simple Skumanich relationship (A. Skumanich 1972), describing the age as a simple power law of the rotation of the star. Recent works on gyrochronology (S. A. Barnes 2003, 2007; E. E. Mamajek & L. A. Hillenbrand 2008; R. Angus et al. 2015, 2019; L. G. Bouma et al. 2023; Y. Lu et al. 2024) reveal a much more complicated picture, with dependencies on the temperature and type of stars. This complex relationship, combining photometric and spectroscopic information, is our motivation to use a multimodal approach for age prediction. In addition, like in binary detection, there are different methods to predict age, and each method is reliable only on a subset of stars. This suggests that combined information might increase the set of predictable stars.

Table 3
Result of Age Prediction Fine-tuning Experiment

Model	Age MAE (Gyr)	Age RMSE (Gyr)
DESA (ours)	**0.61**	**0.94**
MoCo	0.81	1.28
MoCo-clean	0.78	1.23
SimSiam	1.24	1.81
VICReg	0.78	1.23
Spectra only	1.30	1.70
Light curve only	0.80	1.25

Note. Bold values represent the best result.

To create an age data set, we combine two recent age catalogs: L. G. Bouma et al. (2024) and Y. Lu et al. (2024). Although both are gyrochronology methods, they are different. The ages in L. G. Bouma et al. (2024) were calculated using the period and temperature relationship on ages not exceeding 2.7 Gyr. The ages in Y. Lu et al. (2024) also used kinematic information and were calculated over a wider range. We used a combined catalog and crossmatched it with LAMOST spectra. The resulting data set has around 14,000 samples. We used the same setup as in Section 6.3, this time for a regression task, and fine-tuned the model to predict the age and the reported error. The loss function was chosen to be CQR. Table 3 summarizes the results of age predictions. It shows both the MAE and the RMSE of all models. It can be seen that our model outperforms all other models, being the only model with an RMSE lower than 1 Gyr, which was reported by Y. Lu et al. (2024) as the typical error of their method. Prediction plots can be seen in Figure 20 in the Appendix.

7. Conclusions

We presented DESA, a new multimodality model for stellar astrophysics. DESA backbone consists of pretrained unimodality encoders that were trained in a hybrid approach and show state-of-the-art performance. The alignment module, DualFormer, is motivated by the observation that astrophysical data is unique and different compared to common multimodality domains such as vision and NLP. We show that DESA learns a latent space that distills important physical information and can be easily transformed into meaningful diagrams. These diagrams can be used for tasks such as population analysis and outlier detection. The effectiveness of DESA is further demonstrated in various ways, including zero-shot, few-shot, and fine-tuning experiments to predict challenging and important labels such as binarity and age. DESA consistently outperforms all baselines on all experiments with (sometimes very) large margins, proving its superiority in the astronomical domain and marking a new era in data-driven astronomy where multiple measurements are efficiently combined to extract new insights. Each fine-tune task is of great physical importance and deserves an in-depth investigation and separate papers, which we plan for future work. DESA faces two primary limitations: interpretability and single-resolution modeling. Interpretability remains a challenge across ML models, lacking robust methods for results interpretation. Regarding resolution, DESA, while multimodal, was trained on single surveys per modality (Kepler and LAMOST), without explicit consideration for survey-specific resolutions. The last point is related to the applicability of DESA with other surveys. In general, DESA can be applied to different pairs of surveys, such as APOGEE (Abdurro'uf et al. 2022) and TESS (G. R. Ricker et al. 2014), straightforwardly, since nothing in the suggested method itself is specific to LAMOST and Kepler. While the preprocessing is also general, there might be some practical adaptations needed in that part. For example, TESS samples are much shorter compared to Kepler. This means that it might be more challenging to crop each sample into a representative view (in that case, overlapping views can be used). Another example is that the ASPCAP pipeline in APOGEE already moves the wavelength into the rest frame, which makes this step redundant in the spectra preprocessing. However, extending DESA with multiple surveys of the same modality (e.g., APOGEE, LAMOST, TESS, and Kepler together) would require encoding the wavelength and cadence of each survey, which is one of our future directions. Addressing interpretability and multiresolution capabilities stands as our main goal for future research. Nonetheless, DESA represents a potential groundbreaking advancement in multimodal stellar astronomy.

Moreover, beyond its architectural innovations, DESA achieves consistently superior performance across diverse tasks, including zero-shot stellar classification, few-shot regression of photometric properties, and fine-tuning for physically complex problems like binary detection and stellar age inference. In particular, DESA distinguishes itself by recovering classical diagrams (e.g., HR and CMD) from raw embeddings and separating physically degenerate populations—such as synchronized binaries and young stars—without requiring external labels. These results demonstrate that DESA is not merely a predictive model, but a foundation model capable of extracting physically meaningful structure from heterogeneous data. We anticipate that DESA will serve as a powerful framework for future data-driven discovery in large stellar surveys, facilitating population studies, anomaly detection, and improved parameter estimation across the HR diagram.

Acknowledgments

This research was partially supported by the Israeli Science Foundation grant 1834/24.

Appendix
Supplementary Graphs

In the following Appendix we show Figures 14–20 which are referenced in the main text.

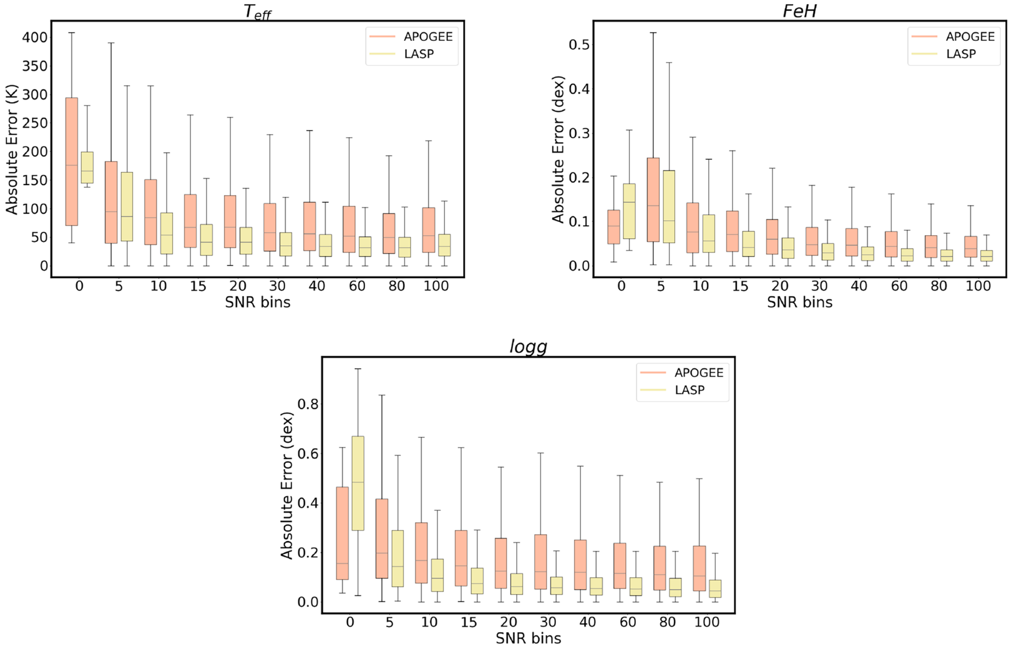

Figure 14. Box plots of MAE vs. SNR for the APOGEE test set and the LAMOST test set.

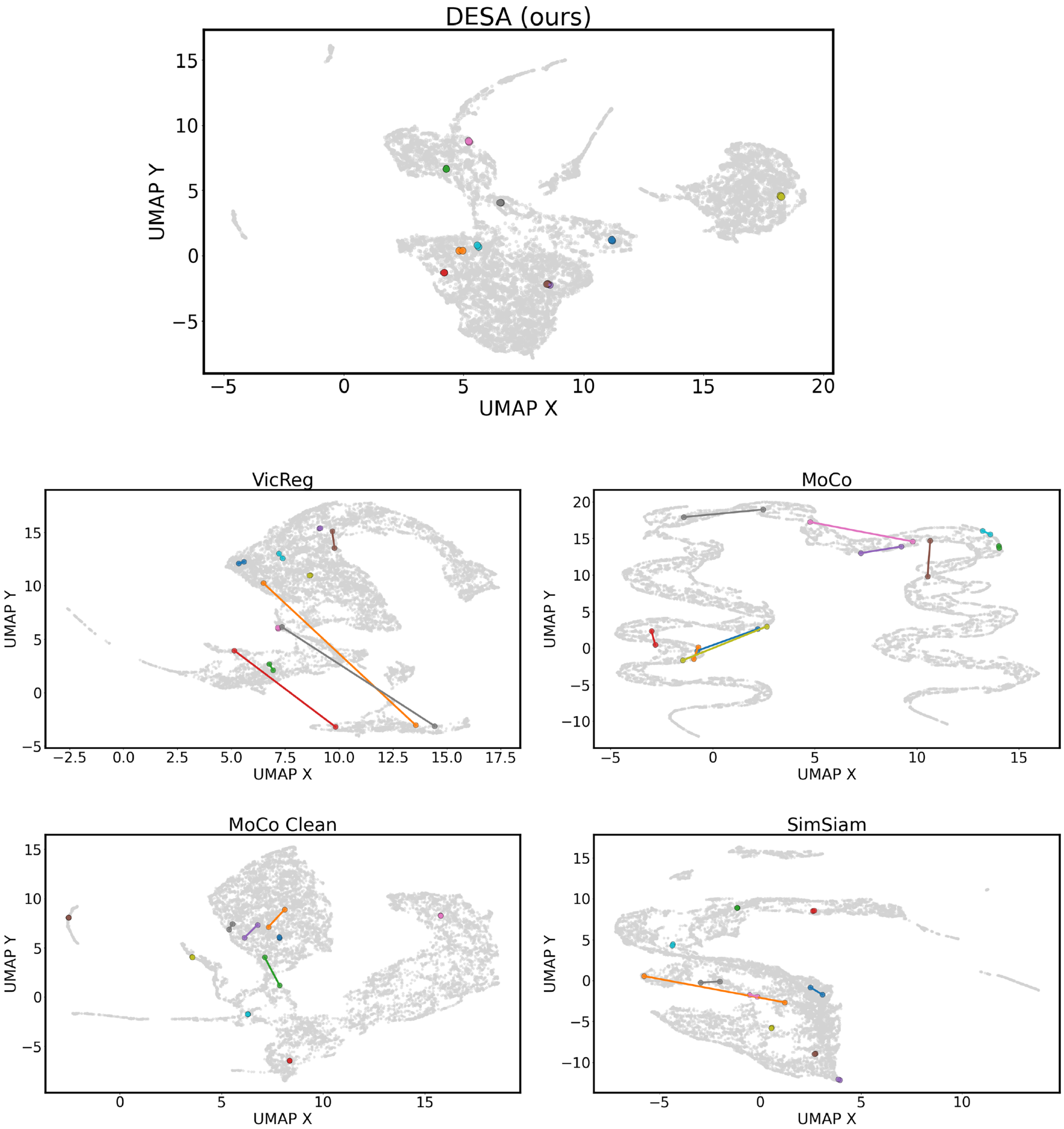

Figure 15. UMAP of all models. All points are in shaded gray, and 10 stars with pairs of spectra are colored on top. For each pair, all samples are colored with the same color and connected by a line. We can see the visual difference between DESA (upper figure) and all other models—in DESA, pairs arrange very close to each other, implying that they have very similar embeddings. In other models, there are samples of the same star that are very distant, which implies very different embeddings.

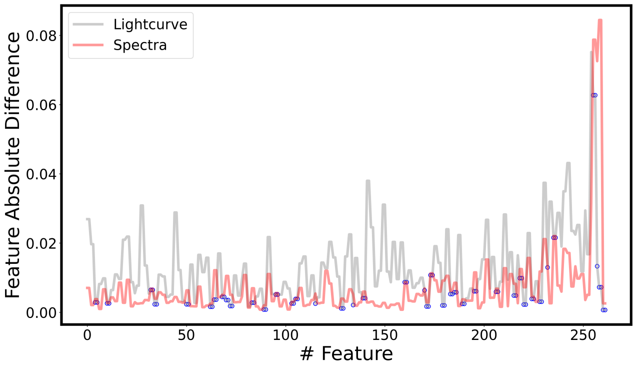

Figure 16. Feature sensitivity study. The *x*-axis shows the feature index in the final features of DESA on the test set. The *y*-axis shows the difference between the original features and features that were generated with only spectral input (red) or only light-curve input (gray). The blue circles mark indices where the light-curve-only features are closer to the original features compared to the spectra-only features.

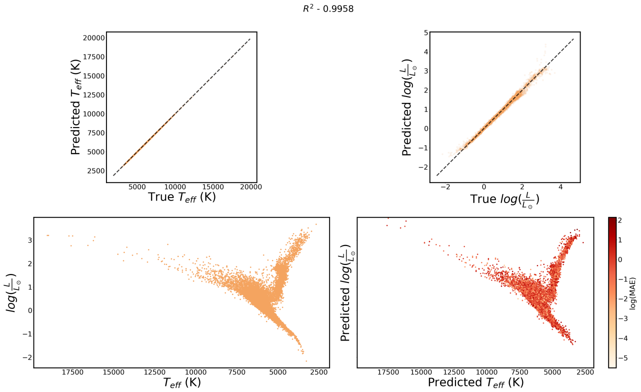

Figure 17. Upper panel—few-shot learning of $T_{\rm eff}$ and $\log(L/L_\odot)$ using linear regression on 20% of the test set. Lower panel—the resulting HR diagram (right) and true HR diagram (left). Colors proportional to the MAE.

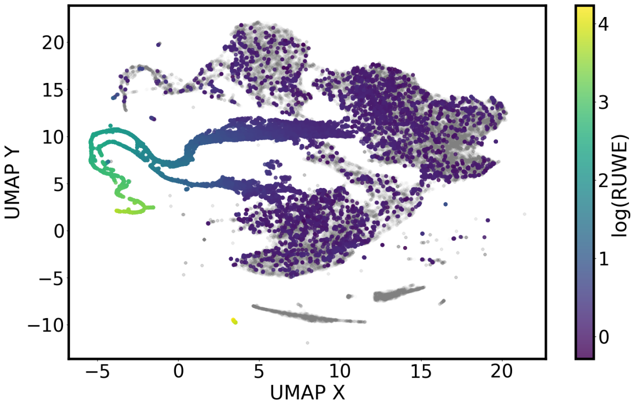

Figure 18. UMAP of DESA final embeddings. All points are in gray, and points with RUWE > 1.4 are colored according to their RUWE value.

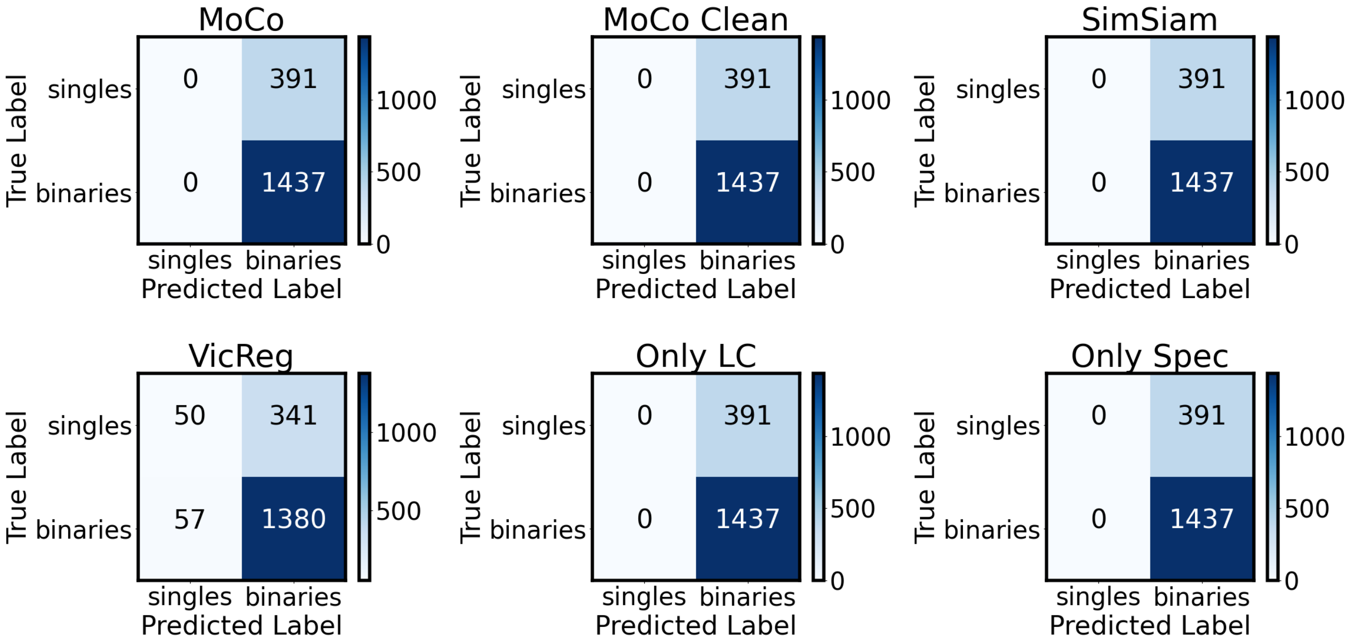

Figure 19. Confusion matrix of alternative models on a binary prediction task.

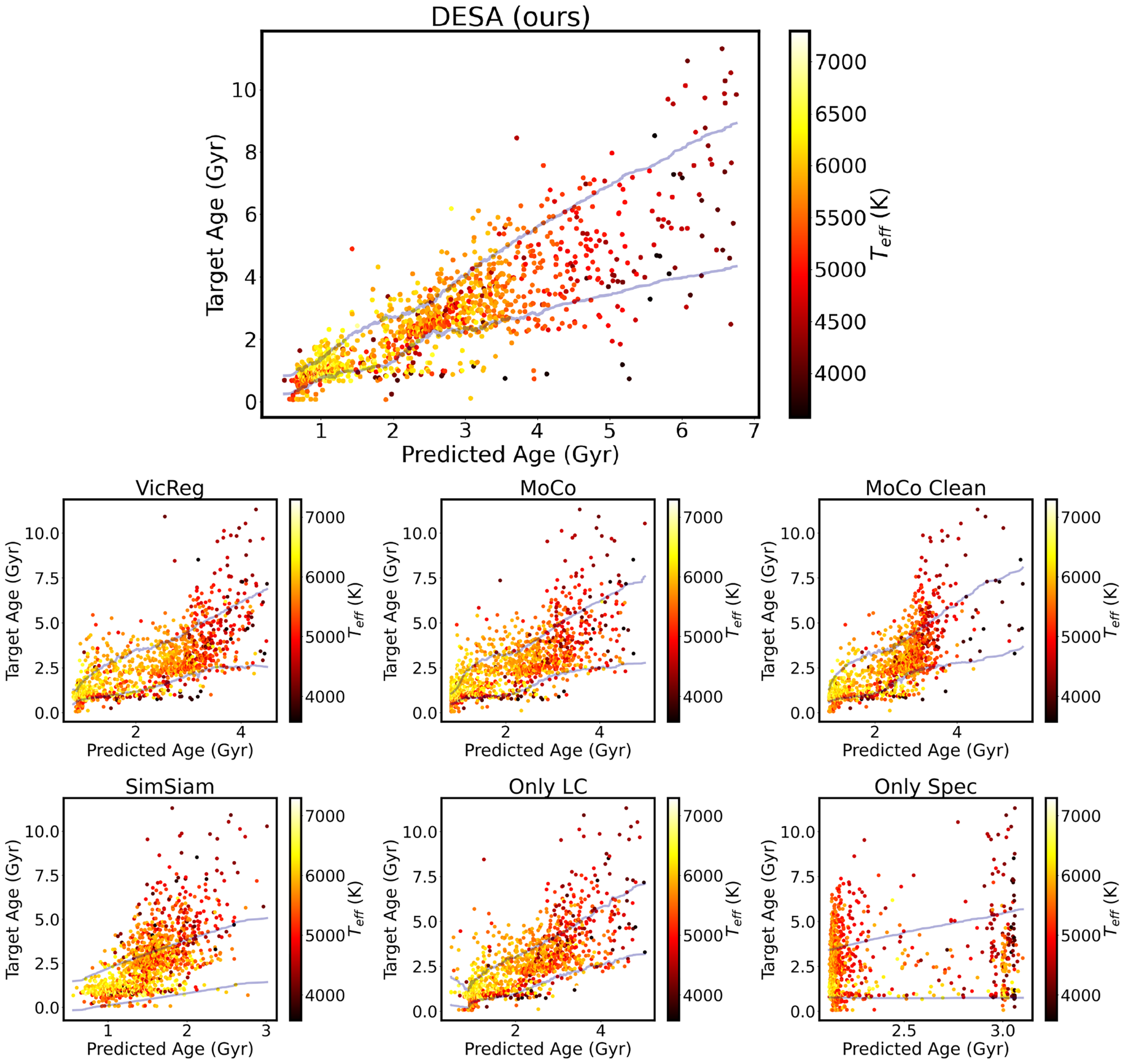

Figure 20. Results of age prediction. The dark blue lines represent a 72% confidence interval. Colors represent $T_{\rm eff}$.

ORCID iDs

Ilay Kamai https://orcid.org/0009-0008-5080-496X
Alex M. Bronstein https://orcid.org/0000-0001-9699-8730
Hagai B. Perets https://orcid.org/0000-0002-5004-199X

References

Abdurro'uf, Accetta, K., Aerts, C., et al. 2022, ApJS, 259, 35
Angus, R., Aigrain, S., Foreman-Mackey, D., & McQuillan, A. 2015, MNRAS, 450, 1787
Angus, R., Morton, T. D., Foreman-Mackey, D., et al. 2019, AJ, 158, 173
Bai, Y., Liu, J., Wang, Y., & Wang, S. 2020, AJ, 159, 84
Bailer-Jones, C. A. L. 2000, A&A, 357, 197
Bardes, A., Ponce, J., & LeCun, Y. 2021, arXiv:2105.04906
Barnes, S. A. 2003, ApJ, 586, 464
Barnes, S. A. 2007, ApJ, 669, 1167
Berger, T. A., Huber, D., van Saders, J. L., et al. 2020, AJ, 159, 280
Blancato, K., Ness, M., Huber, D., Lu, Y., & Angus, R. 2020, arXiv:2005.09682
Bouma, L. G., Hillenbrand, L. A., Howard, A. W., et al. 2024, ApJ, 976, 234
Bouma, L. G., Palumbo, E. K., & Hillenbrand, L. A. 2023, ApJL, 947, L3
Brown, C. F., Kazmierski, M. R., Pasquarella, V. J., et al. 2025, arXiv:2507.22291
Chen, D.-C., Yang, J.-Y., Xie, J.-W., et al. 2021, AJ, 162, 100
Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. 2020, arXiv:2002.05709
Chen, X., & He, K. 2020, arXiv:2011.10566
Claytor, Z. R., & Tayar, J. 2025, ApJ, 987, 8
Claytor, Z. R., van Saders, J. L., Cao, L., et al. 2024, ApJ, 962, 47
Claytor, Z. R., van Saders, J. L., Llama, J., et al. 2022, ApJ, 927, 219
Cui, H., Tejada-Lapuerta, A., Brbić, M., et al. 2025, Natur, 640, 623
García Pérez, A. E., Allende Prieto, C., Holtzman, J. A., et al. 2016, AJ, 151, 144
Godoy-Rivera, D., Mathur, S., García, R. A., et al. 2025, A&A, 696, A243
Grill, J.-B., Strub, F., Altché, F., et al. 2020, arXiv:2006.07733
Gulati, A., Qin, J., Chiu, C.-C., et al. 2020, arXiv:2005.08100
Hattori, S., Angus, R., Foreman-Mackey, D., et al. 2025, AJ, 170, 15
He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. 2019, arXiv:1911.05722
Hoffmann, J., Borgeaud, S., Mensch, A., et al. 2022, arXiv:2203.15556
Kamai, I., Bronstein, A., & Perets, H. 2025, Machine-learning Inference of Stellar Properties using Integrated Photometric and Spectroscopic Data, v1, Zenodo, doi:10.5281/zenodo.17088663
Kamai, I., & Perets, H. B. 2025a, AJ, 169, 59
Kamai, I., & Perets, H. B. 2025b, OJAp, 8, 59
Kaplan, J., McCandlish, S., Henighan, T., et al. 2020, arXiv:2001.08361
Koblischke, N., & Bovy, J. 2024, arXiv:2411.04750
Leung, H. W., & Bovy, J. 2019, MNRAS, 483, 3255
Leung, H. W., & Bovy, J. 2023, MNRAS, 527, 1494
Li, G., Lu, Z., Wang, J., & Wang, Z. 2025, arXiv:2502.15300
Li, X., & Lin, B. 2023, MNRAS, 521, 6354
Li, X., Zeng, S., Wang, Z., et al. 2022, MNRAS, 514, 4588
Loshchilov, I., & Hutter, F. 2017, arXiv:1711.05101
Lu, Y., Angus, R., Agüeros, M. A., et al. 2020, AJ, 160, 168

Lu, Y., Angus, R., Foreman-Mackey, D., & Hattori, S. 2024, AJ, 167, 159

Mamajek, E. E., & Hillenbrand, L. A. 2008, ApJ, 687, 1264

Mathur, S., García, R. A., Ballot, J., et al. 2014, A&A, 562, A124

Mathur, S., Huber, D., Batalha, N. M., et al. 2017, ApJS, 229, 30

McInnes, L., Healy, J., & Melville, J. 2018, arXiv:1802.03426

McQuillan, A., Mazeh, T., & Aigrain, S. 2014, ApJS, 211, 24

Morvan, M., Nikolaou, N., Yip, K., & Waldmann, I. 2022, in Proc. of the Thirty-ninth Int. Conf. on Machine Learning: Machine Learning for Astrophysics (ICML), 11

Olney, R., Kounkel, M., Schillinger, C., et al. 2020, AJ, 159, 182

Pan, J.-S., Ting, Y.-S., Huang, Y., Yu, J., & Liu, J.-F. 2024a, arXiv:2405.17156

Pan, J.-S., Ting, Y.-S., & Yu, J. 2024b, MNRAS, 528, 5890

Parker, L., Lanusse, F., Golkar, S., et al. 2024, MNRAS, 531, 4990

Radford, A., Kim, J. W., Hallacy, C., et al. 2021, arXiv:2103.00020

Raghavan, D., McAlister, H. A., Henry, T. J., et al. 2010, ApJS, 190, 1

Reinhold, T., Reiners, A., & Basri, G. 2013, A&A, 560, A4

Reinhold, T., Shapiro, A. I., Solanki, S. K., & Basri, G. 2023, A&A, 678, A24

Ricker, G. R., Winn, J. N., Vanderspek, R., et al. 2014, SPIE, 9143, 914320

Rizhko, M., & Bloom, J. S. 2025, AJ, 170, 28

Romano, Y., Patterson, E., & Candès, E. J. 2019, arXiv:1905.03222

Santos, A. R. G., Breton, S. N., Mathur, S., & García, R. A. 2021, ApJS, 255, 17

Santos, A. R. G., García, R. A., Mathur, S., et al. 2019, ApJS, 244, 21

Santos, Â. R. G., Godoy-Rivera, D., Finley, A. J., et al. 2024, FrASS, 11, 1356379

Skumanich, A. 1972, ApJ, 171, 565

Su, J., Lu, Y., Pan, S., et al. 2021, arXiv:2104.09864

Walmsley, M., Bowles, M., Scaife, A. M. M., et al. 2024, arXiv:2404.02973

Walmsley, M., Slijepcevic, I., Bowles, M. R., & Scaife, A. 2022, in Proc. of the Thirty-ninth Int. Conf. on Machine Learning: Machine Learning for Astrophysics (MLA), 29

Wang, C., Huang, Y., Yuan, H., et al. 2022, ApJS, 259, 51

Wu, Y., Du, B., Luo, A., Zhao, Y., & Yuan, H. 2014, in IAU Symp. 306, Statistical Challenges in 21st Century Cosmology, ed. A. Heavens, J.-L. Starck, & A. Krone-Martins (Cambridge Univ. Press), 340

Zbontar, J., Jing, L., Misra, I., LeCun, Y., & Deny, S. 2021, arXiv:2103.03230

Zhang, G., Helfer, T., Gagliano, A. T., Mishra-Sharma, S., & Ashley Villar, V. 2024, MLS&T, 5, 045069

Zhang, H., Wu, Q., Yan, J., Wipf, D., & Yu, P. S. 2021, arXiv:2106.12484

Zhao, G., Zhao, Y.-H., Chu, Y.-Q., Jing, Y.-P., & Deng, L.-C. 2012, RAA, 12, 723

Zuo, X., Tao, Y., Huang, Y., et al. 2025, arXiv:2504.20290