Old Dominion University
ODU Digital Commons

2023 REYES Proceedings | REYES: Remote Experience for Young Engineers and Scientists

2023

NuDyCLR: Nuclear Dynamic Co-Learned Representations

Víctor Samuel Pérez-Díaz
Center for Astrophysics, Harvard & Smithsonian

Follow this and additional works at: https://digitalcommons.odu.edu/reyes-2023

Part of the Artificial Intelligence and Robotics Commons, and the Nuclear Commons

Repository Citation
Pérez-Díaz, Víctor Samuel, "NuDyCLR: Nuclear Dynamic Co-Learned Representations" (2023). *2023 REYES Proceedings*. 3.
https://digitalcommons.odu.edu/reyes-2023/3

This Paper is brought to you for free and open access by the REYES: Remote Experience for Young Engineers and Scientists at ODU Digital Commons. It has been accepted for inclusion in 2023 REYES Proceedings by an authorized administrator of ODU Digital Commons. For more information, please contact digitalcommons@odu.edu.

NuDyCLR: Nuclear Dynamic Co-Learned Representations

Víctor Samuel Pérez-Díaz[1,2*] and Sokratis Trifinopoulos[3]

[1*]Center for Astrophysics | Harvard & Smithsonian, 60 Garden Street, Cambridge, 02138, MA, USA.
[2]John A. Paulson School of Engineering and Applied Sciences, Harvard University, 150 Western Avenue, Allston, 02134, MA, USA.
[3]Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, 02139, MA, USA.

*Corresponding author(s). E-mail(s): samuelperez.di@gmail.com;

Abstract

NuCLR (Nuclear Co-Learned Representations) is a cutting-edge multi-task deep learning framework designed to predict essential nuclear observables, including binding energies, decay energies, and nuclear charge radii. As part of the REYES Mentorship Program, we investigated the application of dynamic loss weighting to further refine NuCLR's predictive performance. Our findings indicate that while weighting strategies can enhance accuracy in specific tasks, such as binding energy prediction, they may underperform in others. Equal Weighting (EW), the original method employed by NuCLR, demonstrated consistent performance across multiple tasks, affirming its robustness. This report succinctly presents the developments and results of the mentorship program and outlines our anticipation for continued collaboration on this and related projects.

Keywords: deep learning, nuclear observables, multi task learning

1 Introduction

Despite extensive study spanning more than a century, nuclear physics still defies complete quantitative comprehension. Nuclear Co-Learned Representations (NuCLR) [1], a state-of-the-art deep learning model, aims to address this by predicting nuclear observables such as binding and decay energies, and nuclear charge radii. The model achieves such high performance by leveraging multi-task learning. Remarkably, NuCLR exhibits the emergence of essential aspects of the nuclear shell model, such as shell structure, magic numbers, and compliance with the Pauli Exclusion Principle.

To further refine NuCLR's accuracy, various optimization techniques may be employed, allowing exploration of different learning behaviours [2]. NuCLR uses equal weighting for all tasks. Thus, it prompts the question: What alternative combinations of task losses could enhance network performance? Loss weighting techniques arise as a promising avenue to explore, as they provide the means to calibrate or emphasize focus on specific tasks, potentially augmenting the model's precision in predicting key nuclear observables. This work conducts an exploratory analysis of dynamic loss weighting, comparing Equal Weighting (EW), Uncertainty Weights (UW) [3], Dynamic Weight Average (DWA) [4], Loss Balanced Task Weighting (LBTW) [5], and Random Loss Weighting [6]. An extensive review of these and other methods is available in [2, 6, 7].

This report is organized as follows: Section 2 outlines the network architecture used and elaborates on each of the methods compared; Section 3 assesses and contrasts the performances achieved with each method; Section 4 summarizes the key findings, their relevance to NuCLR, and outlines potential future directions. Finally, an appended reflection on the learning experiences gained through the REYES Mentoring Program is included.

2 Methods

In this study, we employ a preliminary version of NuCLR, referred to as *NuCLR without task embedding*. The architecture takes the number of protons and neutrons within a nuclear structure as input. These numbers are first passed through distinct embedding layers, transforming them into dense vectors. The proton and neutron embeddings are then normalized and concatenated. This concatenated vector is fed through a sequence of linear layers interlaced with ReLU activation functions, batch normalization, and dropout for regularization. The output consists of the predicted observables for the given nuclear structure. The embedding weights are uniformly initialized and normalized to ensure a consistent scale.

In our analysis, we explore the following five loss weighting strategies:

2.1 Equal Weighting (EW)

Equal Weighting (EW) ensures a balanced approach by allocating equal weights to each task. When combining the loss functions from different tasks, EW sums them up without any preference or bias towards a particular task. It is the default or *vanilla* method used in the model and in general in NuCLR.

2.2 Uncertainty Weights (UW)

Uncertainty Weights (UW) is a method introduced in [3]. It formulates the multi-task network as a probabilistic model where the loss function is developed by maximizing the likelihood of the observed output. Specifically, for N simultaneous regression tasks, the distribution of the network's output for each task i is modeled as a Gaussian distribution $\mathcal{N}(f_i(x), \sigma_i^2)$, where $f_i(x)$ is the network's output, and σ_i^2 is a learned parameter representing task-dependent uncertainty. The resultant loss function is given by

$$\sum_{i=1}^{N} \frac{1}{2\sigma_i^2} \|y_i - f_i(x)\|^2 + \log \sigma_i, \tag{1}$$

with y_i being the ground truth for task i. The $\log \sigma_i$ serves as a regularization term, so that σ is not increased without limit. This formulation ensures that each task's loss is inversely weighted by its task-dependent uncertainty, meaning that tasks with lower uncertainty receive more weight.

2.3 Dynamic Weight Average (DWA)

Dynamic Weight Average (DWA) , proposed in [4], explicitly calculates a task's loss using a ratio of the current loss to the previous loss. Let $L_i(t)$ represent the loss for task i at timestep t, and N be the total number of tasks. DWA sets the task weights as

$$\lambda_i(t) = \frac{N \exp(r_i(t-1)/T)}{\sum_{j=1}^{N} \exp(r_j(t-1)/T)}, \tag{2}$$

where

$$r_i(t-1) = \frac{L_i(t-1)}{L_i(t-2)}, \tag{3}$$

ant T is a temperature hyperparameter. Essentially, the loss weight vector is computed as a softmax over the ratios of successive loss values from the last two training steps for each task, and this result is then multiplied by the number of tasks. More explicitly

$$\lambda_i(t) = N \cdot \text{softmax}(r_i/T). \tag{4}$$

2.4 Loss Balanced Task Weighting (LBTW)

Loss Balanced Task Weighting (LBTW), introduced by [5], configures the task weight according to the learning speed, characterized by the ratio of the current loss to the initial loss. LBTW's weighting formula is

$$\lambda_i(t) = \left(\frac{L_i(t)}{L_i(0)} \right)^{\alpha}, \tag{5}$$

where α is a chosen hyperparameter. LBTW does not apply a normalization process to the weight values, allowing them to retain their original scale.

2.5 Random Loss Weighting (RLW)

Random Loss Weighting (RLW) [6] considers the loss weights as random variables and samples them from a random distribution in each iteration. The process begins by sampling an unnormalized weight vector from any distribution and then normalizing it into the actual loss weights through a mapping function, such as the softmax function. This ensures the loss weights are non-negative and sum to one. The random nature of RLW allows for a higher probability of escaping local minima, thus potentially improving generalization ability. Empirical evaluations have shown that RLW methods can achieve performance comparable to state-of-the-art methods. It serves as a reasonable baseline for examining the effectiveness of other loss balancing methods.

3 Results

We assess the performance of the various weighting methods discussed earlier, employing the same network architecture and configuration across all experiments. Models are trained for 1000 epochs using a learning rate of 0.01, weight decay of 0.00067, and the entire training data as the batch size. We predict the following nuclear properties: binding energies; charge radius; neutron and proton separation energies; and β energy decay. For LBTW and DWA, specific hyperparameters are set to $\alpha = 0.1$ and $T = 2$, respectively, in line with the original papers [4, 5]. The dataset and other experimental settings remain consistent with the original work, with further details available in [1].

Figure 1 illustrates the validation loss trajectory over the training epochs for each weighting method. All model variations converge rapidly to values on the order of 10^{-2}. Equal Weighting (EW) emerges as the superior method in terms of overall loss during the final epochs, closely followed by DWA and LBTW. Random Loss Weighting (RLW) displays equal or better performance at some epochs but exhibits a more unstable behaviour.

In Figure 2, we present the task-specific Root Mean Square (RMS) error for each weighting approach. This allows us to clearly distinguish the prioritization of tasks by different techniques. Notably, UW and LBTW excel in predicting binding energies, while marginally lagging behind EW in other tasks. Intriguingly, RLW appears to perform the worst in predicting binding energies. This suggests specific strengths and weaknesses associated with each weighting strategy, thus underscoring the importance of method selection based on the desired task focus.

4 Conclusions and future directions

This report provides an exploratory analysis of various dynamic loss weighting techniques as applied to a preliminary version of NuCLR [1]. We made significant progress in understanding both the limitations and opportunities of these techniques. Specifically, Equal Weighting emerges as the most robust strategy, demonstrating balanced performance across multiple tasks. Conversely, Uncertainty Weights, Loss Balanced Task Weighting, and Dynamic Weight Average favor predictions related to binding energies, although they lag behind Equal Weighting in other tasks. These insights suggest that tailored weighting strategies could be instrumental in optimizing task-specific performance. The code used in this report is available at https://github.com/samuelperezdi/nuclr.

While our experiments did not reveal any task weighting method outperforming EW, it is imperative to recognize the limitations inherent in the conducted study. Our

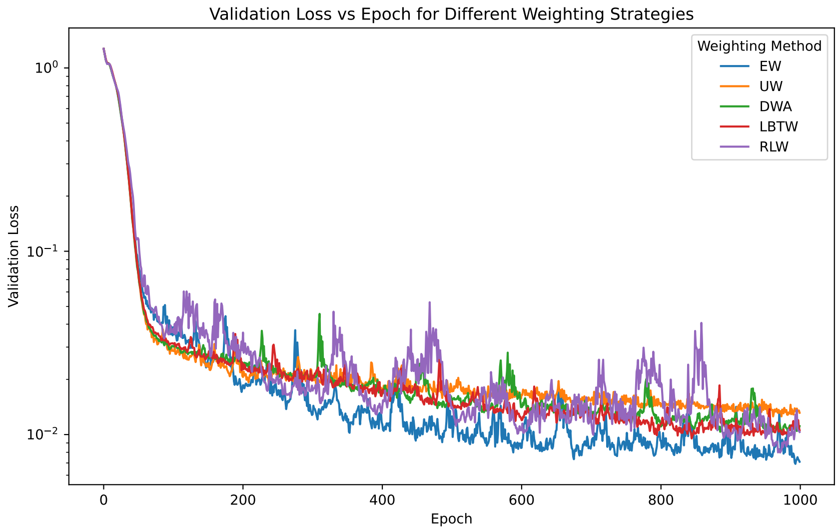

Fig. 1 Validation Loss vs Epoch for each weighting method. We can see that convergence is robust between all techniques. EW stands out with the best overall loss. It is followed closely by RLW and LBTW.

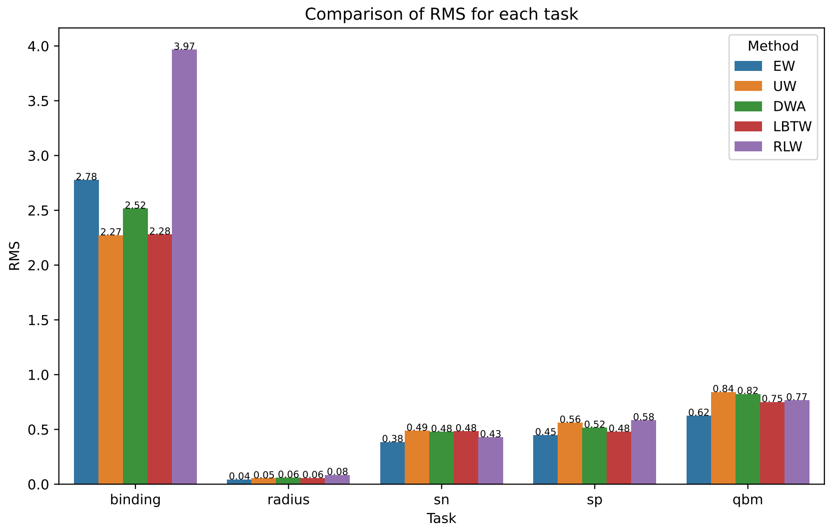

Fig. 2 Task specific RMS for each of the methods. UW and LBTW stand out as the best predicting binding energies.

conclusions stem from a restricted set of runs within a specific network architecture, forming an initial exploration into potential enhancements or refinements for NuCLR. To further unravel the intricacies of how weighting influences learning, future research directions could include:

- Incorporating residuals relative to the semi-empirical Bethe-Weizsäcker mass formula [8, 9] as prediction targets.
- Conducting a hyperparameter sensitivity analysis concerning both network and method parameters (e.g., LBTW: α, DWA: T).
- Exploring alternative weighting strategies, such as IMTL-L [10], GradNorm [11], or GradDrop [12]. Comprehensive reviews of these and related techniques are detailed in [2, 13]. Additionally, `LibMTL` [14], a Python library for deep multi-task learning, offers an extensive collection of implemented state-of-the-art techniques.
- Experiment with additional or different tasks.
- Undertaking a case study using the toy model presented in [1], where the impact of different weighting strategies on learning functions can be directly compared.

The exploration and extension of these paths could lead to a more complete understanding of the interplay between weighting techniques and model performance, fostering the development of more targeted and effective prediction models for nuclear physics.

Learning experience. Participation in the REYES Mentoring Program has afforded the author, Samuel Pérez-Díaz, a profound enrichment of expertise, notably in the realms of nuclear physics and multi-task learning. This new understanding aligns with his goal of using artificial intelligence for scientific discovery and contributes positively to his professional growth. Special thanks are extended to the REYES Team for their work on making this opportunity possible. Special recognition is extended to Sokratis Trifinopoulos, who has provided consistent guidance and camaraderie, which greatly facilitated the author's learning. This experience has fostered excitement for ongoing collaboration with the existing project team and potential future partnerships, all aimed at pushing forward the boundaries of research and exploration.

References

[1] O. Kitouni, N. Nolte, S. Trifinopoulos, S. Kantamneni, M. Williams, Nuclr: Nuclear co-learned representations. arXiv preprint arXiv:2306.06099 (2023)

[2] M. Crawshaw, Multi-task learning with deep neural networks: A survey. arXiv preprint arXiv:2009.09796 (2020)

[3] A. Kendall, Y. Gal, R. Cipolla, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), pp. 7482–7491

[4] S. Liu, E. Johns, A.J. Davison, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2019), pp. 1871–1880

[5] S. Liu, Y. Liang, A. Gitter, in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33 (2019), pp. 9977–9978

[6] B. Lin, Y. Feiyang, Y. Zhang, A closer look at loss weighting in multi-task learning (2021)

[7] T. Gong, T. Lee, C. Stephenson, V. Renduchintala, S. Padhy, A. Ndirango, G. Keskin, O.H. Elibol, A comparison of loss weighting strategies for multi task learning in deep neural networks. IEEE Access **7**, 141627–141632 (2019)

[8] C.v. Weizsäcker, Zur theorie der kernmassen. Zeitschrift für Physik **96**(7-8), 431–458 (1935)

[9] H.A. Bethe, R.F. Bacher, Nuclear physics a. stationary states of nuclei. Reviews of Modern Physics **8**(2), 82 (1936)

[10] L. Liu, Y. Li, Z. Kuang, J. Xue, Y. Chen, W. Yang, Q. Liao, W. Zhang, (iclr, 2021)

[11] Z. Chen, V. Badrinarayanan, C.Y. Lee, A. Rabinovich, in *International conference on machine learning* (PMLR, 2018), pp. 794–803

[12] Z. Chen, J. Ngiam, Y. Huang, T. Luong, H. Kretzschmar, Y. Chai, D. Anguelov, Just pick a sign: Optimizing deep multitask models with gradient sign dropout. Advances in Neural Information Processing Systems **33**, 2039–2050 (2020)

[13] B. Lin, F. Ye, Y. Zhang, I.W. Tsang, Reasonable effectiveness of random weighting: A litmus test for multi-task learning. arXiv preprint arXiv:2111.10603 (2021)

[14] B. Lin, Y. Zhang, Libmtl: A python library for deep multi-task learning. Journal of Machine Learning Research **24**, 1–7 (2023)