NuCLR: Nuclear Co-Learned Representations

Ouail Kitouni* [1] **Niklas Nolte*** [1] **Sokratis Trifinopoulos** [1] **Subhash Kantamneni** [1] **Mike Williams** [1]

Abstract

We introduce Nuclear Co-Learned Representations (NuCLR), a deep learning model that predicts various nuclear observables, including binding and decay energies, and nuclear charge radii. The model is trained using a multi-task approach with shared representations and obtains state-of-the-art performance, achieving levels of precision that are crucial for understanding fundamental phenomena in nuclear (astro)physics. We also report an intriguing finding that the learned representations of NuCLR exhibit the prominent emergence of crucial aspects of the nuclear shell model, namely the shell structure, including the well-known magic numbers, and the Pauli Exclusion Principle. This suggests that the model is capable of capturing the underlying physical principles, and that our approach has the potential to offer valuable insights into nuclear theory.

1. Introduction

The nucleus is the incredibly small and dense region at the center of an atom, consisting of protons and neutrons bound together by the strong nuclear force. Despite having been discovered over a century ago—and the fundamental theory of the strong nuclear force (quantum chromodynamics, QCD) having been discovered 50 years ago—we still lack a precise quantitative understanding of nuclear physics. In principle all nuclear properties are calculable from the theory of QCD, though in practice such calculations are intractable for all but the smallest few nuclei.

Qualitatively, a major breakthrough in understanding nuclei came just after World War II with the development of the *nuclear shell model* by Goeppert-Mayer and Jensen (Nobel Prize in Physics, 1963), which analogous to the atomic shell

model describes the structure of the nucleus in terms of quantum energy states. A key component of this model is the Pauli Exclusion Principle, that no two identical fermions, *e.g.* nucleons (protons or neutrons), can occupy the same quantum state. Since nucleons have spin 1/2, two such particles can occupy the same energy state, one with spin up and the other with spin down, leading to even numbers of protons and neutrons being preferred (more strongly bound). As in the atomic shell model, the most stable states are those with filled shells. Therefore, when adding nucleons to a nucleus, there are *magic numbers* where the binding energy of the next nucleon is substantially less than the previous one, *i.e.* there are configurations that are much more tightly bound due to having a filled shell. The shells for protons and neutrons are filled independently since they are distinguishable particles; however, the number of protons does affect the neutron quantum states and *vice versa*. There are well known magic numbers such as 2, 8, 20, 28, 50, and 82 for both proton and neutron number, Z and N, respectively, and in addition $N = 126$.

A major goal of the modern nuclear physics community is to accurately predict nuclear properties, among which binding energies and charge radii are some of the most important. Traditionally, analytical models have been used, motivated by the known physics—but simplified to make the calculations tractable—and augmented with empirical input to improve their agreement with data (Goriely et al., 2001; Bender et al., 2003; Geng et al., 2005; Möller et al., 2012; Wang et al., 2014).[1] The most accurate of these models, referred to as WS4 (Wang et al., 2014), is able to predict nuclear binding energies and charge radii with a precision of about 300 keV and 0.02 fm, respectively. More recently, machine learning (ML) models have received some attention due to their superior performance potential. Thus far, most ML models were designed for a *single-task learning* (STL) purpose, namely predicting binding energies (Niu & Liang, 2018; Wu et al., 2021; Niu & Liang, 2022; Wu et al., 2022) or charge radii (Utama et al., 2016; Wu et al., 2020; Dong et al., 2022; Ma & Zhang, 2022). An exception is the kernel ridge regression model of (Wu et al., 2022) which utilizes *multi-task learning* (MTL) to simultaneously

---

*Equal contribution [1]Institute for Artificial Intelligence and Fundamental Interactions (IAIFI), MIT, USA. Correspondence to: Mike Williams <mwill@mit.edu>.

[1]With an increase in computational power nuclear *ab initio* methods have recently gained momentum (Navratil et al., 2009; Hagen et al., 2014; Ekström et al., 2015; Novario et al., 2023).
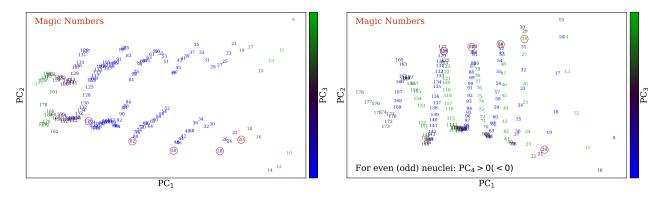
Figure 1. Most important principal components of the neutron embedding representations from (left) early and (right) late in the training. The left panel shows that the most crucial aspects of the nuclear shell model, namely the shell structure, including the magic numbers, and the Pauli Exclusion Principle, arise already early in the training. (The Pauli principle is evident with the even (odd) numbers represented as negative (positve) values of PC2. The ends of the even chains of numbers are the magic numbers where each nuclear shell become full.) In the right panel, the even-odd split now occurs in PC dimension 4, hence is not shown. The shell structure has grown into 3-dimensional spirals, with the largest 4 magic numbers all occurring at local maxima in PC2 and each shell represented as one revolution around an approximately conic surface. Interpreting this spiral structure is ongoing work.

predict both binding and separation energies, achieving a world-leading precision of about $140\,\text{keV}$; *n.b.* this model uses the WS4 predictions as important inputs. ML models have yet to substantially improve upon analytic-model predictions for charge radii. We note that the precision needed for understanding fundamental phenomena in nuclear (astro)physics, are below $100\,\text{keV}$ for binding energies and $0.01\,\text{fm}$ for charge radii;*e.g.* for r-process nucleosynthesis see (Martin et al., 2016; Mumpower et al., 2016).

A major drawback of the previous ML-based approaches is that they lack interpretability. The nuclear shell model is intuitive and surprisingly accurate given its simplicity. We propose an ML-based approach designed to learn a task-independent representation of the nucleus, analogous to the human-learned nuclear shell model picture. To decouple the nuclear representation from task-specific model features, we train to predict a variety of nuclear properties in an MTL setting, all based on the same nuclear representation model, referred to as Nuclear Co-Learned Representations (NuCLR). By exploiting all information available about nuclei, we show that we are able to build a more meaningful model and improve prediction quality that achieves state-of-the-art performance. More importantly, as shown in Fig. 1 we can clearly identify the most important aspects of the nuclear shell model in the latent representations of the model that are shared between all tasks, such as the shell structure, including the well-known magic numbers, and the Pauli Exclusion Principle.

2. More Tasks, More Information

Improving generalization of a prediction task can be achieved in many ways, including by obtaining more or more precise data. However, collecting such data is often expensive, and in the case of nuclear data technologically challenging or even infeasible. Another avenue is exploiting the joint information between different known and measured properties of the nucleus to increase the effective amount of data available for any specific prediction task. This can be achieved by training jointly on all tasks. To exploit data correlations over multiple tasks, a prediction model needs to process data through dependent channels. For neural networks, that most often means sharing weights and layers between tasks. To what degree a model should share processing channels between tasks depends on how similar those tasks are. In the limit of tasks with no joint information, there is no evident benefit to joint training.

In the MTL literature there exist many ways to either manually or automatically control information sharing (Crawshaw, 2020). In order to introduce and motivate choices made for this MTL task, we first show proof of concept with a toy model. The task is to predict the result of the following arithmetic binary functions of two input real numbers: $a+b$, $|a-b|$, $(a+b)^{3/2}$, $\log(a+b+1)$, and $\exp(-\sqrt{a+b}/5)$. The input numbers are treated like symbols or tokens in a language task by embedding them randomly into a high-dimensional space. Once embedded, they lose all their relational properties. The representation vectors are trainable parameters during the learning process that allows the architecture to arrange them in a beneficial way. Schematically, the architecture for this toy-model regression processes inputs is as follows: embed both numbers, concatenate these embeddings, then send the resulting vector through a stack of residual blocks.

In the regime of limited data, we find that using multiple tasks improves generalization on all tasks simultaneously.
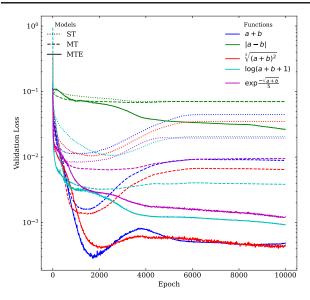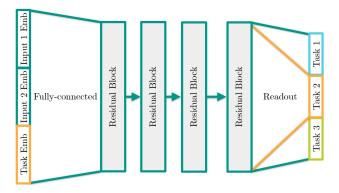
Figure 3. MTE architecture: input-data and task embeddings are concatenated, projected, and passed through a sequence of residual blocks. Different readout heads output predictions for each task. The fact that the same input-data embeddings are used for all tasks encourages structure formation in the embedding space which encodes task-independent information.

Figure 2. For the toy-model arithmetic problem, the multi-task with embeddings (MTE) model generalizes better than the multi task (MT) and single task (ST) models on all tasks in the regime of limited data. Here, only 400 examples for each task are used, 10% for training and the rest for testing.

Figure 2 shows the performance obtained when training on individual targets (single-task, ST) versus joint training on all tasks at once (multi-task, MT), where the latter is superior for all tasks. Here, the MT model has the same layout as the ST models except that the last layer of the model has an output for each task. Note how this architecture hard-codes a high degree of shared computation between tasks. Regardless of the regression target, input numbers undergo the same transformations up to the last layer.

Next, we show that far superior performance is achieved by using a different approach to multi-tasking. Traditionally in MTL, architectures define hyperparameters to control the degree of weight sharing for different tasks. Taking inspiration from language models—which do implicit multi-tasking by conditioning on the task as an additional input—we instead treat the binary operations as trainable embeddings in the same way as the input numbers, and concatenate all three embedding vectors for processing through the model. As in the MT model, the last layer has one output per task. Our approach lets the model decide to which degree tasks are similar and how computations should be shared by moving the task embeddings closer together or farther apart in latent space. Figure 2 shows that this model—labelled MTE, shown schematically in Fig. 3—yields superior performance over both single-target and naive multi-target training.

3. Predicting Nuclear Properties

We now use the MTE approach presented in the previous section to predict nuclear properties. We base our train-

ing on the AME2020 dataset (Wang et al., 2021) for the energy-based quantities and on the compilation of (Angeli & Marinova, 2013) for the charge radii, in both cases using all well-measured nuclei with $Z, N > 8$.[2] The only inputs are three embeddings: proton and neutron number, Z and N, and the task. We choose the following nuclear properties to predict: binding energy per nucleon; charge radius; neutron and proton separation energies, defined as the energy required to remove one neutron or proton from the nucleus; and the energy available for α, β, electron-capture (EC), and β-n decays. All targets are scaled to be contained in the interval [0,1]. In addition, for binding energies we predict the residuals relative to the semi-empirical Bethe-Weizsäcker (BW) mass formula, a direct consequence of the famous liquid-drop model of the nucleus (Weizsacker, 1935; Bethe & Bacher, 1936). The BW formula itself has a prediction precision of about 3 MeV, which is 30 times worse than our target performance. Removing the BW predictions does not greatly affect the prediction performance; however, it does affect the interpretability (discussed later).

When evaluating our models, we must avoid prediction biases such as correlations between the separation energies of a nucleus and the binding energies of its neighboring nuclei, which could be used to artificially obtain excellent performance (Wu et al., 2022); *e.g.*, it is possible to directly obtain (and memorize) the binding energy, $B(Z, N)$, from $B(Z, N) = B(Z, N-1) + S_n(Z, N)$ where S_n denotes the neutron separation energy. Therefore, we train many independent models (100-fold cross-validation), where for each, we withhold 1% of the nuclei from training to be used for validation. In addition, we also remove the separation

---

[2]More details on the dataset, model, and validation procedure are provided in an appendix.
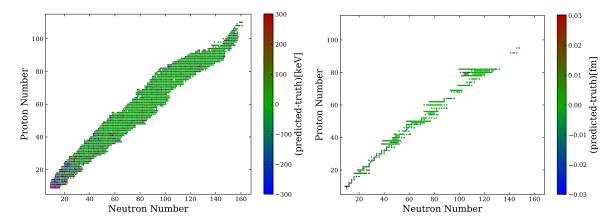
Figure 4. Residuals for NuCLR predictions compared to experimental measurements for (left) binding energy (keV) and (right) charge radius (fm). The vast majority of our predictions are highly precise, especially from the 5th shell up ($N, Z > 28$) where the RMS of our binding energy predictions is only 79 keV *cf.* the binding energies themselves which are $\mathcal{O}(7000) \cdot (Z + N)$ keV.

energies of the neighboring nuclei for each nucleus in a validation sample.

All inputs are embedded into a 1024-dimensional space. The model consists of one linear layer that takes the concatenated embeddings in 3×1024 dimensions and maps them back to 1024 dimensional space, 4 residual blocks with 2 layers of width 1024 each, and one final output layer with one output for each task. We use SiLU activation after each linear layer. This results in a total of $11\,869\,194$ parameters. We train each model for $50\,000$ epochs with learning rate of 0.01, weight decay of 0.01, and a batch size of 4096. The learning rate is cosine-annealed to 10^{-5} during the training.

Figure 4 shows the residuals for our predictions for binding energy and charge radius. The vast majority of our predictions are highly precise, especially for larger nuclei. The root-mean-square (RMS) of the prediction performance of our model on the various tasks is as follows: 130 keV for binding energy; 0.011 fm for charge radius; 130 keV and 140 keV for neutron and proton separation energies; and 139 keV, 180 keV, 179 keV, 196 keV for the energy available for α, β, EC, and β-n decays. We believe that these results establish a new state of the art. In addition, we note that to get competitive performance, previous works considered more human-designed inputs, for instance binding energy predictions of high-precision analytical models (Niu & Liang, 2022; Wu et al., 2022). We could likely improve our results by also including such inputs, especially for smaller nuclei; however, our focus here was to demonstrate the ability to self learn the nuclear shell structure in an interpretable way.

4. Gaining Understanding via Embeddings

The deep learning literature suggests that the crucial phenomenon of *capability emergence* arises as one scales up

the number of parameters, dataset size, and compute across modalities, and in particular, this phenomenon is extremely important in language models (Brown et al., 2020; Wei et al., 2022). A plausible explanation for the performance gains attributed to this phenomenon is the emergence of specialized neural circuits and higher-quality representations. Similarly, since we use the same nuclear embeddings to tackle a range of tasks, we expect our training procedure to lead to the emergence of physically-meaningful representations. Furthermore, if most of the information stored in the embeddings lies on a low-dimensional manifold, then the learned solution could be interpretable, allowing us to gain both confidence in and understanding from the model.

To study the embeddings, we project them down into a low-dimensional space via a principal component analysis (PCA). Figure 1 (left) shows the first two principle components of the neutron embeddings at an early stage in the training. We clearly see that the model has learned to separate the even and odd values of N (vertically on the plot). Recall that N is not given to the machine as an integer, that fact is obscured by providing N as a randomly initialized vector in a 1024-dimensional space. This clear separation of even and odd is a striking statement that it has learned the consequences of the Pauli Exclusion Principle. Furthermore, already at this early stage of training we see *chains* of connected numbers arise with clear breaks between the chains. The break points in the even chains are the well-known magic numbers for N, *i.e.* these break points occur where the nuclear shells become full. Therefore, we conclude that the earliest stages of learning correspond to building embeddings that represent the famous nuclear shell model!

Figure 1 (right) shows the N embeddings at the end of training, where the clear even-odd split due to the Pauli principle remains. Interestingly, we see that the chains of numbers attributed to the shell structure have evolved into

3-dimensional spirals, where the numbers from each shell are roughly equidistant around a circle in the 2-dimensional subspace with the different shells separated along the third PCA dimension. There are many potential explanations for this sturcture, though more detailed study is required to understand this structure. Nevertheless, this spiral structure appears crucial to NuCLR being able to provide state-of-the-art precision for nuclear predictions. Finally, we note that the final embeddings and the observable predictions obtained both with and without the use of the imprecise BW binding-energy predictions as inputs are similar. The primary benefit of using the BW predictions is the clear emergence of the nuclear shell model early in the training.

5. Discussion and Future Directions

NuCLR achieves state-of-the-art performance predicting nuclear observables using a multi-task approach with shared representations. Amazingly, these learned representations exhibit the prominent emergence of the most crucial aspects of the nuclear shell model, suggesting the model is capable of capturing the underlying physical principles. Future directions include more detailed study of the embedding structures, and more generally how to encourage embedding as much information as possible into an interpretable low-dimensional manifold. On the nuclear-physics side, NuCLR could potentially be used to make accurate predictions about many exciting topics in nuclear (astro)physics, including r-process nucleosynthesis (Burbidge et al., 1957), the nuclear neutron skin and its consequences for the structure of neutron stars (Brown, 2000; Horowitz & Piekarewicz, 2001; Gandolfi et al., 2012), the exploration of the boundaries of the nuclear landscape (Erler et al., 2012), and of exotic phenomena such as halo nucleii (Nortershauser et al., 2009) and shape coexistence (Heyde et al., 1983; Wood et al., 1992), and the CKM unitarity puzzle (Seng et al., 2018; Belfatto & Trifinopoulos, 2023; Seng, 2023).

Broader Impact

A generalized view of the method proposed here is that the scientific endeavor itself is a representation-learning problem. Nature presents us with high-dimensional seemingly unorganized data, from which scientists attempt to find low-dimensional representations of the relevant information. Once a low-dimensional representation is found, a model can be built that is not only precise but also trustworthy because we can understand how its predictions arise. In many scientific applications, ML models can easily achieve superior performance but they lack interpretability. Our method re-formulates the ML approach to science in a way that is more aligned with how traditional human-led science works. We showed that if most of the information stored in the embeddings lies on a low-dimensional manifold, then the

ML solution could be interpretable, allowing us to gain both confidence in and understanding from the ML model. In this way, we not only obtain more precise models—but we can potentially learn from the machine. We believe that our approach could be applied to any scientific problem where understanding what the ML model has learned is desired. Finally, we do not foresee any ethical concerns associated with our work.

Appendix: Additional NuCLR Details

Dataset: The dataset is built from the nuclei in the AME2020 collection, which contains both experimentally measured nuclear data and model predictions for as-yet-unobserved nuclei. Only experimentally measured nuclei with $Z, N > 8$ are considered when evaluating model performance; however, predictions for unobserved nuclei are used in the training, which helps in regions where there is little experimental information. In total, about 3k nuclei are used in the training and testing. Note that most nuclei do not have measurements for all of the properties that we predict, *e.g.*, only around one third of the AME2020 nuclei posses a charge-radius measurement. For more details, the AME2020 sample can be inspected interactively at `https://www-nds.iaea.org/relnsd/vcharthtml/VChartHTML.html`.

Model: Figure 3 shows a schematic diagram of the model that takes as input the embedding vectors for proton and neutron number, along with the specific prediction task. During a forward pass, all 3 inputs (proton number, neutron number, task ID) are embedded in a 1024-dimensional space using a learned embedding function. These are then concatenated to form a 3×1024-dimensional vector. This vector is passed through a set of residual blocks of size 1024, ending in one readout linear layer that transforms the 1024-dimensional penultimate activation into a vector with entries for each task. Thus, branching for multi-tasking happens on two levels: (1) the task embedding and (2) the last-layer routing. One data point is defined as (Z, N, Task ID), and the loss for that point considers only the task-relevant output of the network.

Cross-Validation: We use 100-fold cross validation because neighboring nuclei must be excluded in the training (see (Wu et al., 2022), and discussion in the main text above).

For each nucleus in the validation set, all neighbor energies are removed from the training sample, leading to 4–$5\times$ validation set cardinality removed from the training data in each fold. This requires that each validation set be kept small to allow for a large enough training sample. Thus, a large number of folds are needed so that each nucleus appears in at least one validation sample. We note that the previous state-of-the-art model for predicting nuclear properties (Wu et al., 2022) does a much higher number of folds; they train one model per nucleus.

References

Angeli, I. and Marinova, K. P. Table of experimental nuclear ground state charge radii: An update. *Atom. Data Nucl. Data Tabl.*, 99(1):69–95, 2013. doi: 10.1016/j.adt.2011.12.006.

Belfatto, B. and Trifinopoulos, S. The remarkable role of the vector-like quark doublet in the Cabibbo angle and W-mass anomalies. 2 2023.

Bender, M., Heenen, P.-H., and Reinhard, P.-G. Self-consistent mean-field models for nuclear structure. *Rev. Mod. Phys.*, 75:121–180, 2003. doi: 10.1103/RevModPhys.75.121.

Bethe, H. A. and Bacher, R. F. Nuclear Physics A. Stationary States of Nuclei. *Rev. Mod. Phys.*, 8:82–229, 1936. doi: 10.1103/RevModPhys.8.82.

Brown, B. A. Neutron radii in nuclei and the neutron equation of state. *Phys. Rev. Lett.*, 85:5296–5299, 2000. doi: 10.1103/PhysRevLett.85.5296.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Burbidge, M. E., Burbidge, G. R., Fowler, W. A., and Hoyle, F. Synthesis of the elements in stars. *Rev. Mod. Phys.*, 29: 547–650, 1957. doi: 10.1103/RevModPhys.29.547.

Crawshaw, M. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020.

Dong, X.-X., An, R., Lu, J.-X., and Geng, L.-S. Novel Bayesian neural network based approach for nuclear charge radii. *Phys. Rev. C*, 105(1):014308, 2022. doi: 10.1103/PhysRevC.105.014308.

Ekström, A., Jansen, G. R., Wendt, K. A., Hagen, G., Papenbrock, T., Carlsson, B. D., Forssén, C., Hjorth-Jensen, M., Navrátil, P., and Nazarewicz, W. Accurate nuclear radii and binding energies from a chiral interaction. *Phys. Rev. C*, 91(5):051301, 2015. doi: 10.1103/PhysRevC.91.051301.

Erler, J., Birge, N., Kortelainen, M., Nazarewicz, W., Olsen, E., Perhac, A., and Stoitsov, M. V. The limits of the nuclear landscape. *Nature*, 486:509–512, 2012.

Gandolfi, S., Carlson, J., and Reddy, S. The maximum mass and radius of neutron stars and the nuclear symmetry energy. *Phys. Rev. C*, 85:032801, 2012. doi: 10.1103/PhysRevC.85.032801.

Geng, L.-S., Toki, H., and Meng, J. Masses, deformations and charge radii: Nuclear ground-state properties in the relativistic mean field model. *Prog. Theor. Phys.*, 113: 785–800, 2005. doi: 10.1143/PTP.113.785.

Goriely, S., Tondeur, F., and Pearson, J. M. A Hartree-Fock Nuclear Mass Table. *Atom. Data Nucl. Data Tabl.*, 77: 311–381, 2001. doi: 10.1006/adnd.2000.0857.

Hagen, G., Papenbrock, T., Hjorth-Jensen, M., and Dean, D. J. Coupled-cluster computations of atomic nuclei. *Rept. Prog. Phys.*, 77(9):096302, 2014. doi: 10.1088/0034-4885/77/9/096302.

Heyde, K., van Isacker, P., Waroquier, M., Wood, J. L., and Meyer, R. A. Coexistence in odd-mass nuclei. *Phys. Rept.*, 102:291–393, 1983. doi: 10.1016/0370-1573(83)90085-6.

Horowitz, C. J. and Piekarewicz, J. Neutron star structure and the neutron radius of Pb-208. *Phys. Rev. Lett.*, 86: 5647, 2001. doi: 10.1103/PhysRevLett.86.5647.

Ma, J.-Q. and Zhang, Z.-H. Improved phenomenological nuclear charge radius formulae with kernel ridge regression *. *Chin. Phys. C*, 46(7):074105, 2022. doi: 10.1088/1674-1137/ac6154.

Martin, D., Arcones, A., Nazarewicz, W., and Olsen, E. Impact of nuclear mass uncertainties on the r-process. *Phys. Rev. Lett.*, 116(12):121101, 2016. doi: 10.1103/PhysRevLett.116.121101.

Möller, P., Myers, W. D., Sagawa, H., and Yoshida, S. New Finite-Range Droplet Mass Model and Equation-of-State Parameters. *Phys. Rev. Lett.*, 108(5):052501, 2012. doi: 10.1103/PhysRevLett.108.052501.

Mumpower, M. R., Surman, R., McLaughlin, G. C., and Aprahamian, A. The impact of individual nuclear properties on r-process nucleosynthesis. *Prog. Part. Nucl. Phys.*, 86:86–126, 2016. doi: 10.1016/j.ppnp.2015.09.001. [Erratum: Prog.Part.Nucl.Phys. 87, 116–116 (2016)].

Navratil, P., Quaglioni, S., Stetcu, I., and Barrett, B. R. Recent developments in no-core shell-model calculations.

J. Phys. G, 36:083101, 2009. doi: 10.1088/0954-3899/36/8/083101.

Niu, Z. M. and Liang, H. Z. Nuclear mass predictions based on Bayesian neural network approach with pairing and shell effects. *Phys. Lett. B*, 778:48–53, 2018. doi: 10.1016/j.physletb.2018.01.002.

Niu, Z. M. and Liang, H. Z. Nuclear mass predictions with machine learning reaching the accuracy required by r-process studies. *Phys. Rev. C*, 106(2):L021303, 2022. doi: 10.1103/PhysRevC.106.L021303.

Nortershauser, W. et al. Nuclear Charge Radii of Be-7, Be-9, Be-10 and the one-neutron halo nucleus Be-11. *Phys. Rev. Lett.*, 102:062503, 2009. doi: 10.1103/PhysRevLett.102.062503.

Novario, S. J., Lonardoni, D., Gandolfi, S., and Hagen, G. Trends of Neutron Skins and Radii of Mirror Nuclei from First Principles. *Phys. Rev. Lett.*, 130(3):032501, 2023. doi: 10.1103/PhysRevLett.130.032501.

Seng, C.-Y. Model-Independent Determination of Nuclear Weak Form Factors and Implications for Standard Model Precision Tests. *Phys. Rev. Lett.*, 130(15):152501, 2023. doi: 10.1103/PhysRevLett.130.152501.

Seng, C.-Y., Gorchtein, M., Patel, H. H., and Ramsey-Musolf, M. J. Reduced Hadronic Uncertainty in the Determination of V_{ud}. *Phys. Rev. Lett.*, 121(24):241804, 2018. doi: 10.1103/PhysRevLett.121.241804.

Utama, R., Chen, W.-C., and Piekarewicz, J. Nuclear charge radii: Density functional theory meets Bayesian neural networks. *J. Phys. G*, 43(11):114002, 2016. doi: 10.1088/0954-3899/43/11/114002.

Wang, M., Huang, W. J., Kondev, F. G., Audi, G., and Naimi, S. The AME 2020 atomic mass evaluation (II). Tables, graphs and references. *Chin. Phys. C*, 45(3):030003, 2021. doi: 10.1088/1674-1137/abddaf.

Wang, N., Liu, M., Wu, X., and Meng, J. Surface diffuseness correction in global mass formula. *Phys. Lett. B*, 734: 215–219, 2014. doi: 10.1016/j.physletb.2014.05.049.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

Weizsacker, C. F. V. Zur Theorie der Kernmassen. *Z. Phys.*, 96:431–458, 1935. doi: 10.1007/BF01337700.

Wood, J. L., Heyde, K., Nazarewicz, W., Huyse, M., and van Duppen, P. Coexistence in even-mass nuclei. *Phys. Rept.*, 215:101–201, 1992. doi: 10.1016/0370-1573(92) 90095-H.

Wu, D., Bai, C. L., Sagawa, H., and Zhang, H. Q. Calculation of nuclear charge radii with a trained feed-forward neural network. *Phys. Rev. C*, 102(5):054323, 2020. doi: 10.1103/PhysRevC.102.054323.

Wu, X. H., Guo, L. H., and Zhao, P. W. Nuclear masses in extended kernel ridge regression with odd-even effects. *Phys. Lett. B*, 819:136387, 2021. doi: 10.1016/j.physletb.2021.136387.

Wu, X. H., Lu, Y. Y., and Zhao, P. W. Multi-task learning on nuclear masses and separation energies with the kernel ridge regression. *Phys. Lett. B*, 834:137394, 2022. doi: 10.1016/j.physletb.2022.137394.