J of Nuclear Medicine Technology, first published online December 24, 2020 as doi:10.2967/jnmt.120.256628

Intelligent Imaging: Developing a Machine Learning Project

Geoff Currie

School of Dentistry & Health Sciences, Charles Sturt University, Wagga Wagga, Australia

Footline: Developing an ML project

Key words: nuclear medicine, artificial neural network, machine learning, artificial intelligence

Abstract

Artificial intelligence (AI) has rapidly progressed with exciting opportunities that drive enthusiasm for significant projects. A sensible and sustainable approach would be to start building an AI footprint with smaller, machine learning (ML) based initiatives using artificial neural networks (ANN) before progressing to more complex deep learning (DL) approaches using convolutional neural networks (CNN). A number of strategies and examples of entry level projects are outlined, including mock potential projects using CNN to progress toward. The examples provide a narrow snapshot of potential applications designed to inspire readers to think outside the box at problem solving using AI and ML. The simple and resource light ML approaches are ideal for problem solving, accessible starting points for a developing institutional AI program, and provide solutions that can have a significant and immediate impact on practice. A logical approach would be to use ML to examine the problem and identify amongst the broader ML projects which problems are most likely to benefit from a DL approach.

Introduction

Artificial intelligence (AI) has moved at such a fast pace and offers so many exciting opportunities that the inclination is to jump into something big. But a more sensible and sustainable approach would be to start small, learn, build momentum and use internal knowledge to then grow the AI strategy. Individuals or groups eager to build their AI footprint tend to target their efforts to create a world changing deep learning (DL) algorithm. The real challenge is not getting started in AI, it is getting started without bursting the enthusiastic bubble. For example, consider a very keen group of collaborators who wanted to develop a DL algorithm to stratify asthma and silicosis on ventilation lung scans from disease free scans but with no AI experience, limited funding and to be fair, little insight into whether the lung ventilation study had characteristics capable of differentiating the pathologies from normality. A lot of time, energy and money could be wasted before the project failed. Even if the DL provided some stratification, humans will not have learned anything because we do not understand what is happening in the 'magic box'. The first step is to determine whether there is a problem and whether AI has the capacity to be part of the solution. The better approach would use the wealth of information on a large patient series with known outcomes, including radiomic features extracted from the lung scan, and feed those features into a machine learning (ML) based artificial neural network (ANN) and train it based on a binary grounded truth of disease of interest versus no disease of interest. In doing so, the ANN will identify specific inputs or features that individually or collectively provide the highest predictive capability for the disease. It may reveal that the lung scan plays an important role or no role at all. In the case of the former, potential to further automate through DL and convolutional neural network (CNN) might be justified. In the case of the latter, it would avoid the wasted time and money associated with a DL approach. For those considering projects to develop a footprint in AI:

- Consider a problem where the solution provides a good outcome rapidly and use that to build momentum (early win).
- Avoid taking on a high investment (time, resources, energy and funding) project that provides no reward for a long time.

- Ensure the outcome is not trivial. Address a problem that has impact and will generate attention with success.
- Focus your efforts on projects that tangibly connect to nuclear medicine rather than peripheral areas of interest.
- Ensure your team is comprised of credible colleagues (internally and externally).
- Recruit a team comprised of complementary skills associated with enthusiastic colleagues who can accelerate both the project and resulting AI momentum.

Developing an AI strategy for your research team or clinical department is not an overnight journey. Do not expect the exciting project convolved today to have clinical impact any time soon. Consider the scenario where a data rich department that trains an ANN or CNN to perform a task and then validates that algorithm (*1*-*3*). This scenario represents almost all of the current literature on ML and DL in nuclear medicine. This is internally valid data that, with success, provides an algorithm that can reliably perform the prescribed task to enhance internal process and outcomes. Commercialization of these algorithms for more broad impact is a much more difficult process. Firstly, the process of data entry, sharing and management needs to overcome issues of privacy and security (*4*). Secondly, the internal dataset may contain local bias that threatens external validity. This requires rigorous validation with external data. Thirdly, the algorithm may be specific for internal protocols and equipment that does not hold true when parameters or equipment change. For example, the ANN scaling and weighting associated with predicting cardiac events in 123I mIBG heart failure patients (*5*) changes substantially if the uptake phase was undertaken in temperatures that vary by as little as 2 degrees Celsius, the time post injection to scanning time varies, the collimator or energy window varies, and with resolution / sensitivity of specific gamma cameras. Over and above these variations, changing the acquisition or reconstruction parameters will also change scaling and weights. This provides an insight into the discrepancy between the enormous potential applications of ML and DL in nuclear medicine and radiology, and the actual number of commercially available algorithms approved by the US FDA.

Consideration should be given to how the ML output can be neatly integrated into existing graphical output of nuclear medicine scans. This, in part, also speaks to the commercializability of the algorithm. The addition of ML based risk stratification and scoring is a logical step. Consider figure 1 (left) from the context of feasibility and clinical utility of a CNN driven risk score for pulmonary embolism which could be summarized and displayed in a simple format. Perhaps figure 1 (right) is an option where the perfusion SPECT is segmented against the accompanying low-dose CT to predict pulmonary embolism. Segmentation, risk scoring of individual lesions and a total disease burden might be helpful for patients presenting for evaluation of metastatic spread to bone (figure 2). These are all mock-ups for potentially useful AI projects to provoke your own thoughts and ideas. These examples should not constrain your vision, with innovative applications of AI well beyond these simple examples including classification, localization, detection and segmentation projects.

Machine Learning Examples

The following are two simple ML projects using an ANN. This kind of approach can be undertaken with custom development tools or using commercial software packages. A project identified regional analysis of myocardium immediately adjacent to infarcted tissue and the associated 123I mIBG washout as the key predictor of cardiac event in heart failure patients (*6*). This was to provide richer insights using regional SPECT analysis than the previously described global approach to planar analysis on the basis that the pathology cascades from regional to global. The conventional statistical analysis for the outcome was complex, individualized but provided the greatest single predictor of cardiac event and, thus, need for an implantable cardioverter defibrillator (ICD).

The emergence of capability in developing ANN and their use as tools for data analysis in parallel to conventional statistical analysis saw the data re-analyzed. The advantage of the ANN is that it examines all variables in a manner that removes duplication (even if highly correlated) and redundancy, weights variables to optimize the predictive outcome and in doing so identifies the variable or combination of variables that provides the greatest predictive power. The ANN identified the same single best predictor as did conventional statistical analysis (washout from tissue immediately adjacent to the infarcted tissue) but also determined that the combination of 2 very simple variables provided highest predictive capability (figure 3, top) (*6*). In this case, the ANN was not DL or CNN but rather a ML tool designed to analyze more critically the relationships between variables in relation to predicting the outcome (cardiac event).

The ANN architecture initially comprised of 84 scaling layer inputs, 3 hidden layers of 6 nodes each using a logistic activation function for the binary output layer. The weighted squared error method was used to determine the loss index and the neural parameters norm was used for the regularisation method. A Quasi-Newton training method was adopted and the architecture optimized at 2 inputs (change in LVEF and planar washout), 2 hidden layers of 6 and 1 node respectively, and a binary output. This was evaluated using receiver operator characteristics (ROC) analysis demonstrating an area under the curve of 0.75 which correlated to a sensitivity of 100% and specificity of 50%. The ANN indicated that the best predictor of a cardiac event is the coexistence of a decrease in the LVEF of 10% or greater and a planar washout of 30% or greater.

A second example was part of a detailed analysis of metrics to characterize injection kinetics in PET using topical sensors (*7*). The commercial software provided a number of metrics and scores for interpretation by users. While individual metrics provide useful tools, combining the errors associated with each produced an overall larger error. There were 45 input variables in 863 patients using a binary classification of dose extravasation or no dose extravasation. The heat map / correlation matrix identified a number of redundant variables and the highest correlation scores associated with tc50 (0.838), ndAvgN (0.749), and CEnd ratio (0.721); consistent with the conventional statistical analysis. A 60:20:20 instances split was used for training, selection and testing. The ANN architecture included 3 hidden layers of 5, 5 and 1 nodes respectively, using a logistic activation function for a binary output. As with the first example, the weighted squared error method was used to determine the loss index, the neural parameters norm was used for the regularisation method, and a Quasi-Newton training method was employed. The final architecture of the ANN reflects the optimized subset of inputs with the lowest selection loss, in this case, 4 inputs, 3 hidden layers of 3, 4 and 1 node respectively, and a binary output (figure 3, bottom). It is worth noting that the purpose of removing redundancy and optimising both the number of inputs and the complexity (number and depth of nodes) is to minimise over- or under-fitting. This was evaluated with ROC analysis that demonstrated an area under the curve of 1.0 correlating with a sensitivity of 100% and specificity of 100% although binary classification testing revealed a less than perfect predictive model with classification accuracy of 0.75.

Code, Frameworks and Libraries

The tools for AI (figure 4) can be generally classified as datasets (serving the apatite of DL for data), architecture (functional model), framework (to enhance functionality), hardware (CPU, GPU etc), libraries (sources of datasets, architectures and frameworks), and code (language). Python (https://www.python.org/doc/) is a high level open source, fairly simple and intuitive programming language that can be used for direct coding in ML and DL or the source code for Keras based network development (application training interface). It is not the only programming language used for CNN development but is perhaps the most common and certainly the most accessible and usable for those early developers. NumPy (https://numpy.org/) is an extension package for Python that allows algebraic functions, Fourier transformation, integration of Python code with other types of code (Fortran and C for example). NumPy is like adding the scientific calculator to the standard calculator, enhancing Pythons scientific functionality.

Keras (https://keras.io/) is a deep learning library that runs as an application programming interface (API) for python in conjunction with TensorFlow. PyTorch (https://pytorch.org/) is an alternative open source library of tools for training phase platforms. TensorFlow (https://www.tensorflow.org/) is a graphics-based computation platform that stores data in objects (tensor) rather than integers or strings. It is used as a backend platform for

CNN deployment but can include Keras for broader application in the training phase providing an end-to-end platform. The framework and associated libraries are described as the ecosystem tools while the architecture and datasets are the landscape. Caffe (https://caffe.berkeleyvision.org/) is the University of California, Berkley DL framework and is useful for those a little more adventurous with DL research. KNIME (https://www.knime.com/) is a DL analytics platform for CNN development, training and implementation. KNIME integrates with the Keras DL architecture and the TensorFlow ML library. Theano (http://deeplearning.net/software/theano/) is a similar library for scientific extension of Python as NumPy with additional features associated with speed and optimization. Deeplearning4j (https://deeplearning4j.org/) is a Java based, open source DL library that can be integrated with Keras as the Python API. Pandas (https://pandas.pydata.org/) are powerful data analysis toolkits for Python; essential a library of tools for those using Python for data analysis.

Convolution Neural Networks

The CNN provides a rich range of opportunities for development of AI projects. This includes CNNs for auto-segmentation of complex datasets, disease detection and segmentation, image reconstruction methods, development of pseudo-CT for attenuation correction of PET/MRI or PET alone, scatter correction and de-noising of images, and radiation dosimetry on serial PET images. These approaches could adopt a variety of CNN architectures (eg. AlexNet or VGGNet), encoder/decoder architectures (eg. UNet) and generative adversarial networks. An entry level CNN project might avoid the complexity of dynamic data or tomographic data. Instead, a simple CNN architecture (eg. AlexNet) could be used either using direct python coding or via commercial software (eg. Deep Learning Toolbox from MATLAB - https://au.mathworks.com/products/deep-learning-hdl.html) to run a simple classification project on planar images. Consider a simple import of a lung perfusion scan into a CNN trained against known cases of pulmonary embolism or otherwise. The CNN, once trained and validated, could triage perfusion scans with findings consistent with pulmonary embolism for more urgent reporting. This avoids the greater complexity of comparing ventilation studies because the role is to triage, not to diagnose (eg. the graphical abstract).

Summary/Conclusion

AI and ML have almost infinite applications in nuclear medicine and medical imaging. The examples above are far from exhaustive. They provide a narrow snapshot of potential applications designed to inspire readers to think outside the box at problem solving using AI and ML. The key consideration is identifying a problem that has tangible benefits to be solved and identifying the appropriate tool to solve it. In many cases, problems do not need AI, ML or DL to solve. Frequently, the simpler and less resource intensive ML approaches are ideal for problem solving. Indeed, the plethora of ML based applications in nuclear medicine and medical imaging are not only accessible starting points for a developing institutional AI program, but also provide solutions that can have a significant and immediate impact on practice and patient outcomes. Nonetheless, ML applications are often overlooked in favor of the higher order, perhaps more "prestigious" DL approaches that demand higher levels of expertise, time and resources. A logical approach would be to use ML to examine the problem and identify amongst the broader ML projects which problems are most likely to benefit from a DL approach.

References

1. Currie G, Hawk KE, Rohren E, Vial A, Klein R. Machine learning and deep learning in medical imaging: intelligent imaging, *J Med Imaging Radiat Sci*. 2019;50: 477-487.

2. Currie G. Intelligent Imaging: anatomy of machine learning and deep learning, *J Nucl Med Technol*. 2019;47: 273-281.

3. Currie G. Intelligent imaging: artificial intelligence augmented nuclear medicine, *J Nucl Med Technol*. 2019;47: 217-222.

4. Currie G, Hawk KE, Rohren E. Ethical principles for the application of artificial intelligence (AI) in nuclear medicine and molecular imaging, *Eur J Nucl Med Mol Imaging*. 2020;47: 748-752.

5. Iqbal B, Currie G, Kiat H. Regional cardiac sympathetic nervous system evaluation using 123I-mIBG SPECT in patients with heart failure, *J Med Imaging Radiat Sci*. 2018;49: 397-405.

6. Currie G, Iqbal B, Kiat, H. Intelligent imaging: radiomics and artificial neural networks in heart failure, *J Med Imaging Radiat Sci*. 2019;50: 571-574.

7. Currie G, Sanchez S. Topical sensor metrics for 18F-FDG positron emission tomography dose extravasation, *Radiography*, 2020; ahead of print: DOI:https://doi.org/10.1016/j.radi.2020.07.013.

List of Figures

Figure 1: On the left is a mock summary output for a CNN based risk algorithm for pulmonary embolism using ventilation and perfusion mismatch. On the right is a mock summary output for a CNN based risk algorithm for pulmonary embolism using low-dose CT and perfusion SPECT mismatch. The coronal and sagittal slices represent 2 different patients; one with mismatch consistent with a high likelihood of pulmonary embolism (coronal) and the other matching defect associated with lower likelihood of pulmonary embolism (sagittal).

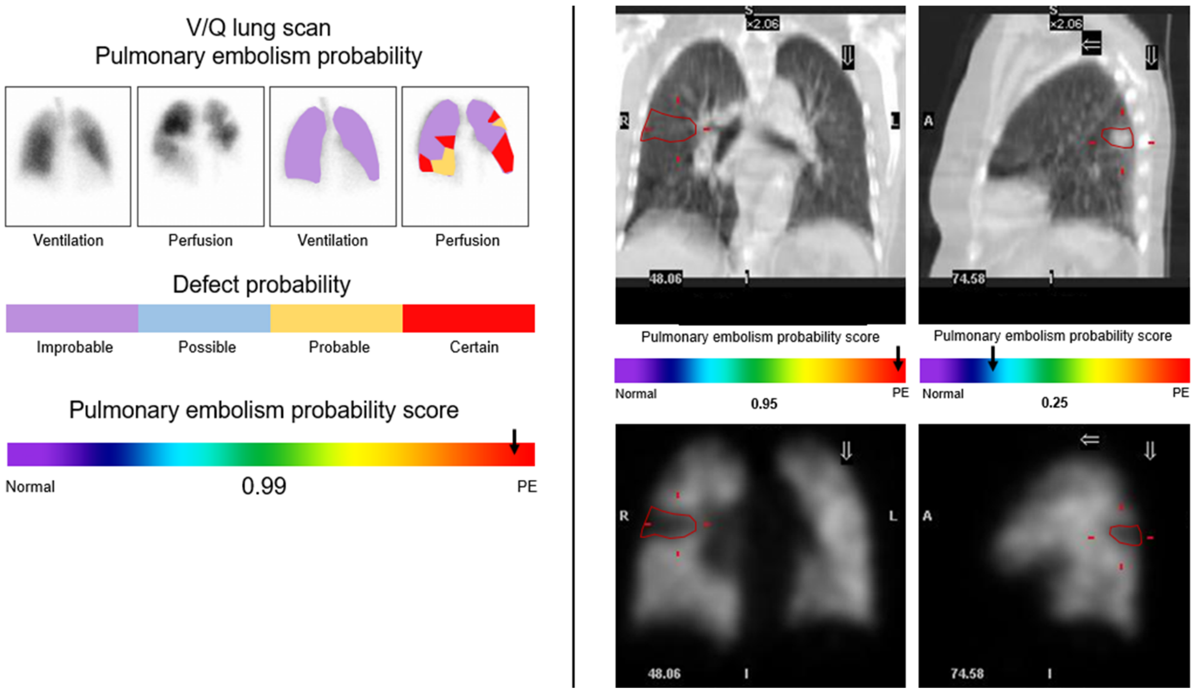


Figure 2: Mock summary output for a CNN based risk algorithm for skeletal metastases with probability classification for various outcomes and risk assessment for individual lesions.

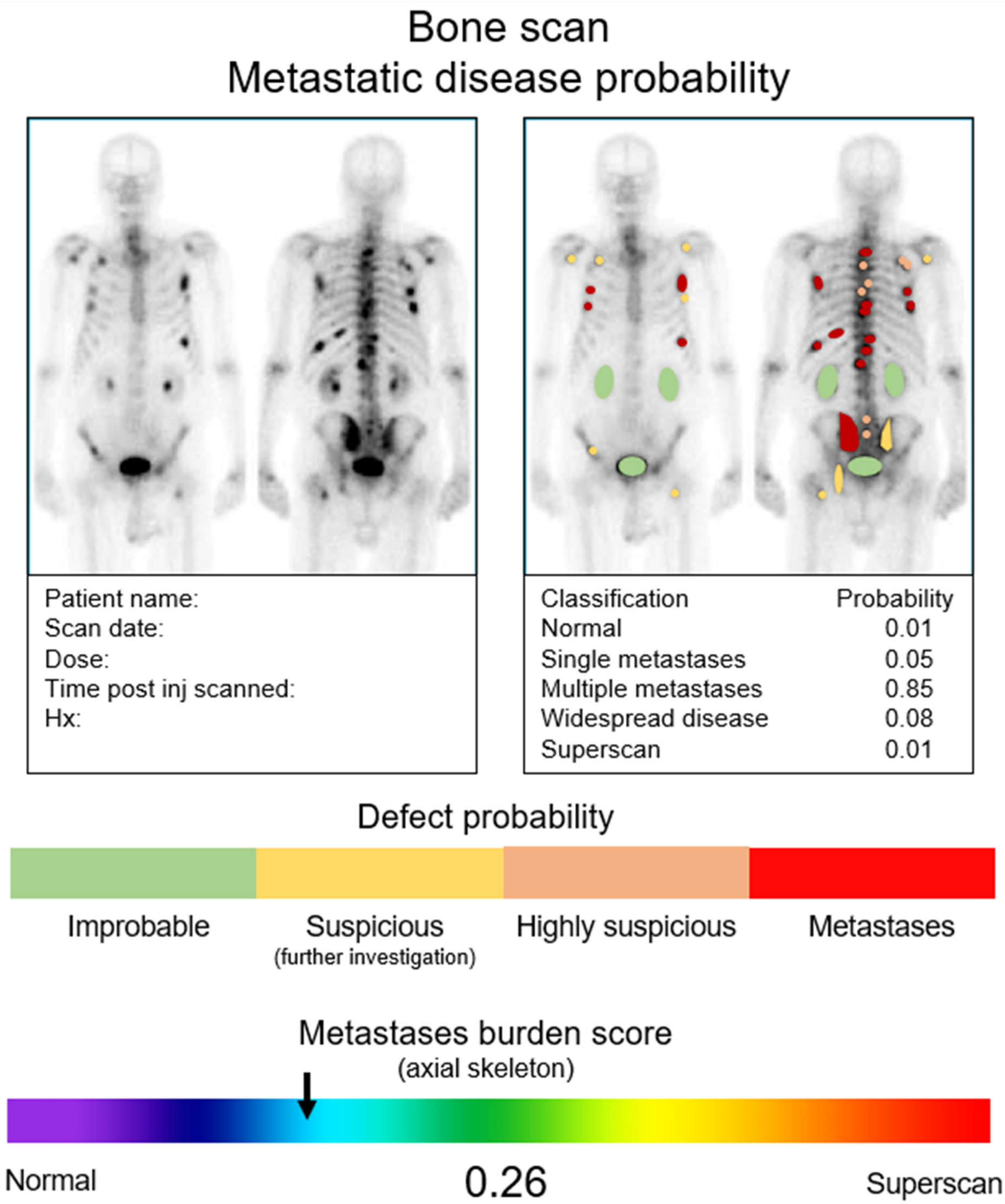
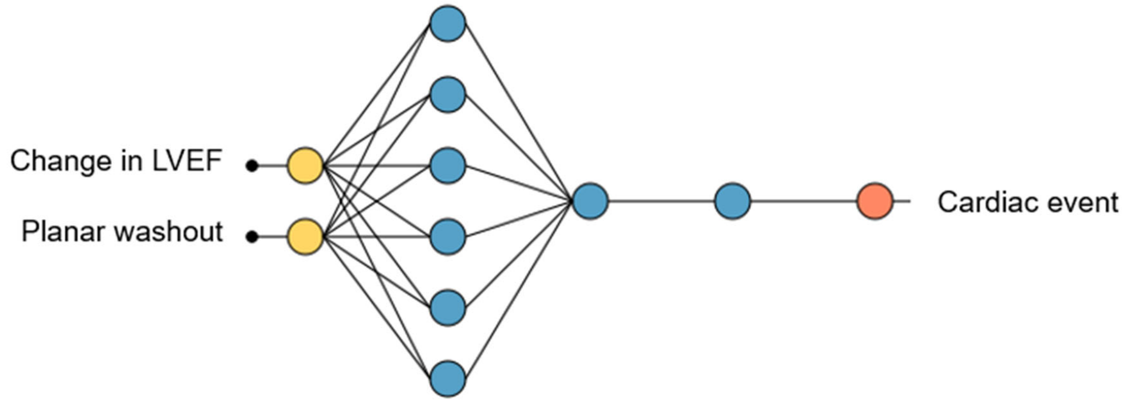


Figure 3: The final ANN architecture optimizes inputs and network complexity for the heart failure cohort (top) and the dose extravasation cohort (bottom). Here, the number of inputs, and the number and complexity of nodes are optimized.



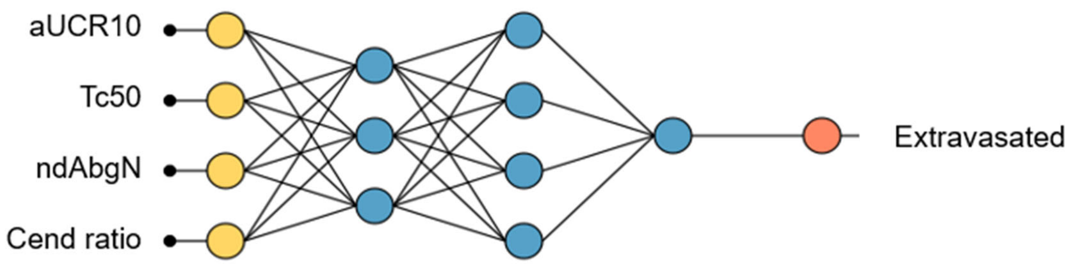


Figure 4: Schematic representation of the relationships between hardware, code, frameworks, architectures, libraries, datasets and the deep learning solution. The DL landscape is defined by two orange zones and the DL ecosystem by the purple and yellow zones.