Yin and Yang, The Harmony of AI and Domain Knowledge

Ruiyuan(Ralph) Kang

1. 1Introduction

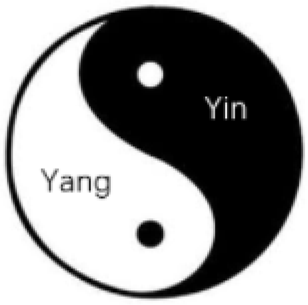
The wide range of applications of machine learning/artificial intelligence in various fields needs no further explanation, and its power is undeniable. However, for scientific research, there are still some unignorable shortcomings of existing mainstream AI.

A. The tradeoff of Accuracy and Truth
Todays popular machine learning techniques are ultimately designed to simulate the inner workings of a system by fitting observational data to it. However, many questions can be put forth: does the observed data represent the true operating law? How much noise or uncertainty is there and how much sampling bias is there? Does the model learn the natural law or the misinterpreted scripture? This involves a trade-off between truth and accuracy (generalizability vs. accuracy). It is true that we can train a complex model to even capture trivial perturbations, but it may not perform well on the test set (Overfitting), and even with pruning, early stopping and regularization techniques, it is still difficult to generalize the model to similar dataset (transfer learning is a "do as Romans do" story). Many studies play such tricks similar to this example: "Machine learning-based prediction of the taste of coffee at around 10:00 am shaken by Tony, a Starbucks waiter " However, the model is not accurate at 3:00 pm (because Tony is tired), and itis even less accurate for James. This is the embarrassment of most machine learning models, especially supervised learning models, that they are all pursuing accuracy but not about natural law, however, what scientists want is the model can simulate the real world. In the book of the Elements of Statistical Learning, this is called the tradeoff of Bias and Variance.

B. black-box models are far from enough
Most existing machine learning models are black-box models, and deep learning is especially black. Although they are powerful, they are far from adequate for scientific research, because we want to get insights from them that we can understand, explain, and extend to other aspects (which is not the same as EXplainable Artificial Intelligence, XAI). For example, admittedly, we can train a machine learning model to predict how an apple will smash into Newton, but from the model, we will never know the law of gravity, but the simple law of gravity has been generalized to infinite applications, as the old saying goes: a picture is worth a thousand words (to the inventor of vision transformer, "an image is worth 16*16 words" [1]). Similarly, some people also say that "a model is worth a thousand datasets" [2]. I agree with this saying from the example about Newton. For scientific discovery, explicit formulas/laws are indeed what we need,

not just functional implementations of AI.

Obviously, machine learning is not a silver bullet, and the understanding of natural laws that have underpinned the development of human society to this point is an even more valuable asset. Not as many peoples opinion, to my perspective, the zero -sum game should not be the relationship between machine learning and physical knowledge. According to the ancient Chinese philosophy theory of Taiji, I think the two should be the relationship of Yin and Yang. These two obviously different and even contrary methods can actually be complementary, interconnected, and interdependent. Yin means the darkness, coldness, implicit, which can represent Artificial intelligence, Yang means bright, hotness, explicit, which can represent Domain Knowledge. As the Chinese saying goes, one system cannot develop well without the force from both Yin and Yang. Artificial intelligence and Domain Knowledge should complement and support each other. So how to do this? I think we can split the story into two parts and further four pieces to discuss.



2. AI dominating

2.1. Domain-Knowledge-assisted AI (Yin assisted by Yang)

There are three main steps of the machine learning model: feature engineering/data pre-processing, model construction, and data post-processing.

The traditional machine learning model cannot be separated from feature engineering, and there is even the saying "10% of machine learning, 90% of feature engineering", and later there is also the joke of "The amount of artificiality will generate the corresponding amount of intelligence". Although there are fixed paradigms for feature selection in feature engineering, such as embedding method, filter method, the feature extraction is various. If one does not know the physics of the problem, the common practice is to transform data (the use of various kernel functions), PCA, time-domain or frequency-domain statistical features (such as various moments, FFT energy), which is basically at the mercy of God, because if these features are not good, one does not know where is the direction to improve. Domain Knowledge can provide guidance for feature engineering, for example, Mel-frequency cepstral coefficients(MFCC), which is commonly used in the field of speech recognition, can split audio signals into multiple valuable features. Although the popular end-to-end deep learning greatly reduces the necessity of feature engineering and lowers the threshold of machine learning, the process of data augmentation more or less embeds the researchers understanding of the invariance and the physics of problem, e.g., the choice of whether RGB, grayscale or binary images will be used depends on whether the author believes that color, pixel intensity or only the geometry in the image is important, respectively.

A common practice on model construction is to work on the loss function. A machine learning model is an approximate function of the natural law of a system, and the degree of approximation depends on the adequacy of the constraints, which are reflected in the construction of the loss function. Common models learned are tensor-to-tensor mapping, which can actually be seen as a generalized algebraic equation, however, common physical systems are mostly differential equations, and simple zero-order algebraic term constraints are of course not enough, so auxiliary loss functions should be constructed through the conversion of differential equation balance. For example, the construction of the loss function for the first-order heat conduction equation is as follows.

$$q_x = a\frac{dT}{dx} \Rightarrow Loss = q_x - a\frac{dT}{dx}$$

This is a simple example, but more complex ones are also of the same reasoning. Such embedded higher-order constraints are now widely used in PDE, ODE approximations [3], such as replacing explicit Navier-Stokes solvers [4]. Of course, there can be other types of constraints as well which can be learned from optimization books.

In addition, selecting suitable kernel functions or algorithms from domain knowledge on the distribution of the physical system is also popular. A simple example is that if only the neighborhood plays a decisive role in a physical system, then a simple radial basis function network or K-Nearest Neighbors algorithm may give better results than Multilayer Perceptron (MLP). There is an old saying that one takes the behavior of ones company, graph neural networks are often used to infer local correlations and similarities, which you can see their ghosts from the "people you may know" recommendations of Twitter and Facebook.

Of course, the existence of pre-processing implies the existence of post-processing. You can use the model to find the Nussle number, or you can use it to find a bunch of parameters and then derive them into the Nussle number, but of course, the Nussle number is more essential and more generalized. What output is obtained from the network and what it is transformed into is definitely the earliest formulation based on human physical insights, from this point of view, post-processing is actually the very first pre-processing.

2.2. Domain Knowledge-discovered AI (Yin generates Yang)

How can we domain Knowledge from AI? I think there are two main methodologies at the moment. One is to still use black-box models of machine learning, giving it the ability to discover new knowledge, which in essence is no different from mainstream functional AI; the other is to let machine learning generate explicit models (equations, formulas) describing the essence of the physical systems. Since these models have the theoretical ability to discover new theories/mechanisms, so some people also call such models Machine Scientist [5].

The former we have seen in much big news and breakthrough scientific studies, such as Alphafold of DeepMind to directly solve the protein-folding problem; such as applying the Bayesian optimization algorithm to discover new materials, and similar plays using generative models, graph neural networks. In these applications, we dont know how these models work, but we know they can work. The question remains on whether what is learned is accurate or real? This has to be verified by, for example, experiments with new materials guided by AI, luckily, these black-box models are more efficient and reliable than human scientists at the time being in most cases, so they are still desirable.

The latter case can be unified into the domain of symbolic regression [6]. Traditional symbolic regression is to learn the explicit formula between inputs and outputs given possible inputs (such as mass, acceleration) and possible operations (such as addition, subtraction, multiplication, division). E.g., if the output is force, then a search by symbolic regression can learn that the relationship between mass and acceleration should be multiplicative. Since how symbols are chosen is not a continuous and differentiable process, in general, such studies mostly use evolutionary algorithms such as genetic algorithms. In a way, symbol learning is a kind of search with direction and skill (violent cracking).

While symbolic regression is exciting, it also has the hard problem that the operational complexity of the actual formula (the number of operations m), and the sum of the number of input terms and operation types of the formula (which I call the number of bases, n) determine its search space to be n^m, so that an increase in complexity and number of bases will bring about an exponential increase in the search space. Although this limits the application of symbolic regression to complex formula discovery, think about the fact that it took Kepler 4 years to fit the formula approximating the orbit of Mars, symbolic regression still has merit. In order to solve the problem of complex knowledge discovery, several solutions have been given by the academic community till now.

1, Sparse Regression (SR) [7], in which you have already known what the possible input terms are, and then these terms form a linear relationship (only subtraction and multiplication), then the complex problem is converted into a linear regression problem.

For example, an energy conservation equation that considers heat transfer is as follows:

$$-\frac{\partial q_x}{\partial x}dx-\frac{\partial q_y}{\partial y}dy-\frac{\partial q_z}{\partial z}dz+\dot{q}\,dx\,dy\,dz=\rho c_p\frac{\partial T}{\partial t}dx\,dy\,dz$$

Although each term connected by addition and subtraction is a nonlinear term when each term is considered as a whole, for example

$$set X_1 = \frac{\partial q_x}{\partial x} dx$$

Equivalence analogy, then the whole equation becomes a linear equation. One may ask, how do you know what terms are there? This question either depends on your expertise, for example, you know what terms there are in the N-S equation and all you need is to calculate the coefficients, or you can build a “world model” that includes all the terms you know, for example, some people take thousands of common mathematical terms from Wikipedia and build a so-called "alphabet". However, SR does also have some imperfect points. First, some complex models cannot be disassembled into fairly simple independent terms in advance, such as the Nusser number representation in laminar liquid film condensation (from Dr.Tiejun Zhangs slides).

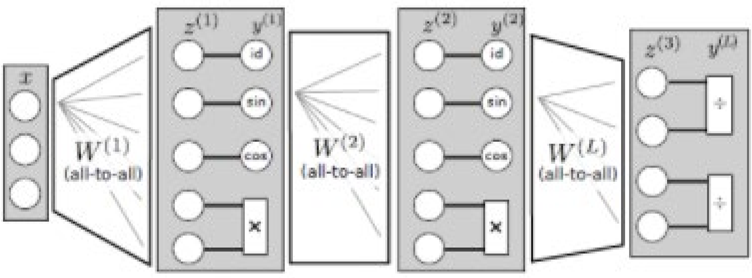
$$\overline{Nu} = \frac{\bar{h}L}{k_l} = \underbrace{\frac{4}{3\sqrt{2}}}_{0.9428}\left[\frac{\rho_l \cdot (\rho_l - \rho_v) g \cdot h'_{fg} L^3}{k_l \cdot \mu_l \cdot (T_{sat} - T_w)}\right]^{1/4}$$

Second, do we get the real formula or its explicit approximate solution(e.g., Taylor expansion of the real formula), if it is an approximate solution, then we may still not get proper physical insights from generated formulae, so what is the essential difference with the black box model? Some more advanced sparse regression algorithms use network-type structures, such as EQL [8], symNet [9], all these can be seen as a multi-layer nesting of single-level sparse regression models. That is,

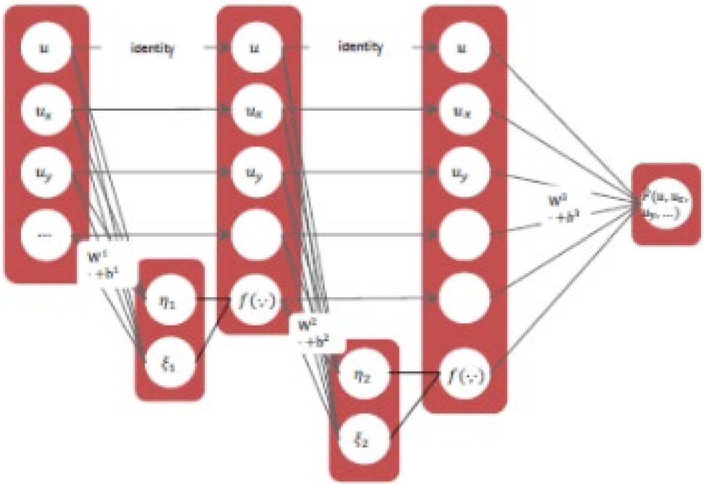
$$y = f(g(x)) \; EQL$$

$$y = f(x, g(x)) \; Symnet$$

However, these complex algorithms still have the common shortcomings as the simpler ones.



Architecture of EQL



Architecture of symNet

2, Decouple into subproblem. n^m is a large search space, but if n can be split into n_1 and n_2, and m can be split into m_1 and m_2, then the potential search space would be much smaller. That is,

The essence of this is the relationship of breaking a large system into several smaller subsystems, similar to a multi-layer SR.

So the question is how to split it reasonably? Cranmer Mills of Princeton [10] uses the use of graph neural networks because graph neural networks can be viewed from both graph and node layers, he even took this to study dark matter, COOL! But not all systems can be represented in graph form, so the method is pretty limited.

3, Learn from human scientists [11].
Scientists have many techniques for dealing with complex mathematical problems, such as dimension(unit) analysis, generalization from simple problems to complex problems (similar ideas are also common in curriculum learning, reinforcement learning, deep Boltzmann machines, etc.), and by gradually narrowing the search space through such a pass, the difficulty of symbolic regression can be greatly reduced.

3. Domain Knowledge dominating

Since each discipline has its characteristics and way to process problems (that is why call them domain knowledge), I personally know very little about this, just put here to lead.

3.1. AI-assisted Domain Knowledge Analysis (Yang assisted by Yin)

The most common way is the coupling of a machine learning model and a physics-based model to produce a hybrid model, which is a special kind of ensemble learning to some extent. For example, for a system, the main estimate is generated by the physics-based model and the machine learning model only learns the difference between the prediction of the physical model and the actual measurement (learning uncertainty) [12], This kind of idea is similar to the Boosting approach in Ensemble learning, or Resnet in deep learning. In this type of model, the physical model is responsible for determining the direction and the machine learning model is used to improve the accuracy. Since historically there are multiple physical models for the same problem, it is possible to engage a machine learning model to adjust the weight of each physical model for various conditions. In this way, this model is more like the blending method in AI, the machine learning model is the second layer of the meta learner.

Another way to play is to use machine learning to determine which physical terms are more influential, which is called sensitivity analysis in physics, thus simplifying the analysis process; in machine learning, this is a usage of feature selection.

3.2. AI-discovered Domain Knowledge (Yang generates Yin)

I left this section open for the sake of symmetry, but this section is really self-explanatory because, throughout the history of AI development, numerous AI operations have come from the insights of Domain Knowledge. For example, the earliest Perceptron model was derived from the working mechanism of neurons, and then the layer convolution Layer actually benefited from the way humans observe things from the local to the total. The terms Momentum, Cross-Entropy, and so on will be familiar to anyone who has studied mechanics, informatics, and thermodynamics.

4. Concluding remarks

In this blog, I just mention some methods to couple AI and Domain Knowledge, and of course, this world is more fascinating and colorful than what I have written, and it is totally unnecessary to consider the partitions of domain knowledge and AI in one application. What I want to deliver is both tools are necessary for scientific research, just like Yin and Yang in Taiji. The worrying trend now is blind belief in AI in all aspects of the world. I personally believe that the current functional AI is like a bait for fishermen (or perhaps a proton put by the trio), if we humans rely on the current mainstream functional AI and gradually give up the pursuit of true knowledge, despite our lives will be greatly facilitated, but our knowledge of the world will stop to grow. Smart people should learn to play Taiji, lend force (AI) to fight force (physical problems), and use the so-called new paradigm of science - AI to promote the further development of human society.

Reference

[1] A. Dosovitskiy et al., “An Image is Worth 16x16 Words:Transformers for Image Recognition at Scale,” Oct. 2020.
[2] C. Rackauckas et al., “Universal Differential Equationsfor Scientific Machine Learning,” pp. 1–55, 2020.
[3] M. Raissi, P. Perdikaris, and G. E. Karniadakis,“Physics-informed neural networks: A deep learning framework forsolving forward and inverse problems involving nonlinear partialdifferential equations,” J. Comput. Phys., vol. 378, pp.686–707, 2019,

doi: 10.1016/j.jcp.2018.10.045.
[4] M. Raissi and G. E. Karniadakis, “Hidden physics models:Machine learning of nonlinear partial differential equations,” J.Comput. Phys., vol. 357, pp. 125–141, 2018, doi:10.1016/j.jcp.2017.11.039.
[5] R. Guimerà et al., “A Bayesian machine scientist to aidin the solution of challenging scientific problems,” Sci. Adv.,vol. 6, no. 5, 2020, doi: 10.1126/sciadv.aav6971.
[6] M. Schmidt and H. Lipson, “Distilling free-form natural lawsfrom experimental data,” Science (80-.)., vol. 324, no.5923, pp. 81–85, 2009, doi: 10.1126/science.1165893.
[7] S. Rudy, A. Alla, S. L. Brunton, and N. A. Jun, “Data-drivenidentification of parametric partial differential equations,” pp.1–17, 2018.
[8] G. Martius and C. H. Lampert, “Extrapolation and learningequations,” 5th Int. Conf. Learn. Represent. ICLR 2017 - Work.Track Proc., no. 1, pp. 1–13, 2019.
[9] Z. Long, Y. Lu, and B. Dong, “PDE-Net 2.0: Learning PDEs fromdata with a numeric-symbolic hybrid deep network,” J. Comput.Phys., p. 108925, 2019, doi: 10.1016/j.jcp.2019.108925.
[10] M. Cranmer et al., “Discovering symbolic models fromdeep learning with inductive biases,” Adv. Neural Inf. Process.Syst., vol. 2020-Decem, no. NeurIPS, pp. 1–25, 2020.
[11] S. M. Udrescu and M. Tegmark, “AI Feynman: A physics-inspiredmethod for symbolic regression,” Sci. Adv., vol. 6, no. 16,2020, doi: 10.1126/sciadv.aay2631.
[12] Y. Yin et al., “Augmenting Physical Models with DeepNetworks for Complex Dynamics Forecasting,” 2020.