Prospero's Books: A Distributed Architecture for AI

Invited Extended Abstract

Bhubaneswar (Bud) Mishra(✉)

Courant Institute, NYU, New York City, USA
mishra@nyu.edu

Abstract. This preliminary note and its sequels present a distributed architecture for AI (Artificial Intelligence) based on a novel market microstructure. The underlying game theory is based on Information-Asymmetric (Signaling) games, where deception is tamed by costly signaling. The signaling, in order to remain honest (e.g., *separating*), may involve crypto-tokens and distributed ledgers. Here, we will present a rough sketch of the architecture and the protocols it involves. Mathematical and computational analyses will appear in the subsequent sequels.

To my Recommenders and Verifiers: *Rekha's* and *Vera's*....

> *But release me from my bands – With the help of your good hands.*
> *Gentle breath of yours my sails – Must fill, or else my project fails,*
> *Which was to please. Now I want – Spirits to enforce, art to enchant;*
> – Prospero in Shakespeare's **The Tempest**

1 Minsky's Society of Mind

In 1986, in his book *Society of Mind*, AI pioneer Marvin Minsky wrote: "What magical trick makes us intelligent? The trick is that there is no trick. The power of intelligence stems from our vast diversity, not from any single, perfect principle." However, it has remained unclear how and whence these artificial agents congregate to form such a society of mind. Nonetheless, it has been argued that should such a society emerge from artificial computational building blocks, it will possess a great power as it would view "a mind as a society of agents, as opposed to the consequence of some basic principle or some simple formal system,... different agents can be based on different types of processes with different purposes, ways of representing knowledge, and methods for producing results."

We propose a game theoretic approach to establish and maintain such a society where the agents signal, interact strategically and maintain stable separating Nash equilibrium. In particular, we will also introduce two sets of non-strategic agents: namely, *Recommenders* and *Verifiers*, whose interactions are crucial for the system to reach and maintain"good" (honest, separating, desirable) Nash equilibria.

D. V. Hung and M. D'Souza (Eds.): ICDCIT 2020, LNCS 11969, pp. 41–49, 2020.
https://doi.org/10.1007/978-3-030-36987-3_3

The paper is meant to be widely accessible - primarily to computer, data, intelligence and finance engineers - and hence deeper mathematical treatment is relegated to subsequent sequels.

2 Turing's Artificial Intelligence

Classically, intelligence and its role in problem solving have been difficult to formalize. While computability has a widely-accepted model in terms of *Church-Turing thesis*, *Turing-reducibility* and *Turing-universality*, as a consequence of these, it remains impossible to define AI by its general problem solving capability, since there remain many useful and natural decision problems whose undecidability is straightforward: the classical decision problem represented by the Halting Problem or, equivalently, demonstrating computational equivalence of two "programs." In other words, given two programs: one genuine and other (presumably) imitative, there can be no decision procedure to determine if they are Turing equivalent. These statement have deep implications on how we may want to define Artificial Intelligence.

The solution Turing suggested was in terms of an Information-Asymmetric Signaling games: involving certain set of sender agents, some of which will have the type Oracles (e.g., humans) and the others the type Imitators (e.g., models). The senders send certain signals (e.g., conversational statements in English) to receivers (e.g., humans) who must act strategically by only responding to Oracles, while ignoring Imitators. Such a game may be called an *Imitation (Signaling) Game* and the receiver's test a *Turing Test.* Similarly, by also assigning types to receivers (i.e., Oracles and Imitators), one may extend the Imitation Game to also include *Reverse Turing Tests.* As a signaling game the classical Imitation Game and its extension both have Nash Equilibria: some trivial as Babbling or Pooling but others far more relevant to present discussion: namely, *separating.* A natural way to define Artificial Intelligence would be in terms of Imitators' ability to achieve a reasonably informative and stable pooling (non-separating) Nash Equilibrium when introduced into a society of human Oracles. In other words, the receiver must respond in exactly the same manner independent of whether the sender is an Oracle or Imitator.

Our approach involves extending the system to include additional non-strategic agents: namely, Recommenders and Verifiers. They will have no explicit utilities to optimize (or even, satisfies) other than those described in terms of betting and winning (or losing) certain tokens. These tokens may be implemented in terms of a cryptographic object, which must be "hard" to counterfeit (replicate or manufacture); transactions among recommenders and verifiers can be verified in terms of various local and global properties (expressed in terms of a model-checkable logic, e.g. propositional temporal logic), assuming that a non-tamperable model (e.g., a Kripke structure) is dynamically created by some other agents, who are additionally required to employ costly-signaling. While it will remain unspecified as to how the Recommenders and Verifiers and their models may be constructed and deployed, it is conjectured that as long as they satisfy certain system-wide distributed liveness (Recommender's responsibilities) and safety (Verifier's responsibilities) conditions, the game should result in and maintain a stable Nash equilibrium that will also be pooling (Imitators are indistinguishable from Oracles). The intuitions supporting this conjecture is out-of-scope of this introductory note. Since the work presented here combines various ingredients from Imitation Game, Society of Mind, Signaling Games, Turing Learning, Generative-Adversarial Networks, Approximate Bayesian Computation, Bayes and Empirical Bayes techniques and Causality Analysis, we will only briefly comment on these connections here, leaving more details to the full paper and additional sequels.

3 Sketch of an Architecture for AI

The architecture involves

1. Multiple AI modules ("Models" M's) in an ensemble working on multiple data sets ("Domains" D's). Such a set will be referred to as an "*Ecosystem.*" There will be some effort to eliminate over-fitted models by using an empirical Bayes approach to control false-discovery rate in multiple hypotheses testing (described later in this paper).
2. *Ranking:* All models are assumed to be generative; the generated data can be compared to future data in order to provide a rank function ("Rank(M, D)").
3. *Oracle(s):* It is assumed that there exists a model M^* (perhaps not yet discovered) that on a data set D performs exactly, without any error. Namely, there is a distance function such that

$$\text{Distance}(D(M^*), D) = 0,$$

   where $D(M^*)$ is the data generated by M^*. Thus Rank(M^*, D) is superior to Rank(M, D) for any M in the ecosystem.
4. *Goal:* Use a recommender-verifier system in a signaling game to identify the best approximation to oracle M^* for a domain D.

As discussed earlier, we assume a set of agents who participate in the game to rank a model M (either an existing one, combination of existing ones or a new one). They will be referred to as Recommenders and Verifiers.

A recommender agent selects a domain and a data-set D associated with the domain and a model M. The recommender may publish hypotheses in support of M (e.g., why (M, D) may best approximate M^*, for instance, using qualitative reasoning, causal support, past history or new computational analysis on real or synthetic data.). The recommender stakes some utilities (e.g., tokens). If the recommended model is deemed to be ineligible for a competition the recommender loses the stake. The recommender also publishes an estimated rank Rank$_R(M, D)$.

One or many verifier-agents provide their estimated ranks of the model: Rank$_V(M, D)$.

A model eligible for competition may then be set up to test if the true rank Rank(M, D) is above or below the median of the ranks estimated by the verifiers. If the competition occurs and the rank is above (resp. below) the median, then half of the agents who estimated a rank above (resp. below) the median win and the other half of the agents who estimated a rank below (resp. above) the median lose. All models found eligible for competition are included in the ecosystem, together with its computed rank (or its recommended rank, if no verifier challenges it).

Losers pay the winners a predetermined amount of utilities (e.g., tokens).

Suppose a recommender devises a strong Oracle-like model. He is then incentivized to contribute the model to the ecosystem as he is sure that it will be eligible for a competition and most likely attract sufficiently many verifiers (resulting in no loss of stake); he also expects a win from the verifiers who will underestimate the power of the model.

As a side effect, over time, weak recommenders whose models do not lead to competitions can get pruned out.

Note that independent of the result of the competition an Oracle-like model always gets evaluated and included in the ecosystem (if and only if it is eligible for competition). There may be further opportunities to earn rent from the future use of the model in the ecosystem.

A good recommender must avoid contributing weak random variations of an oracle model once it has been achieved, while the domain is stationary. In this case most strong verifiers will bet against him and win.

A recommender is also incentivized to work on a domain where the models can be further improved (instead of investing in a domain that already has an Oracle-model or a strong approximation to it.) This situation may arise as a result of the nonstationarity of data.

A weak recommender is deterred by the fact that his recommendations will not be eligible for competition and will result in loss of stake. In addition, just introducing black boxes without any reasoning (or domain-specific prior), may attract strong verifiers who will bet against him.

Similarly, a weak verifier will not be able to accumulate utilities as he will face more frequent losses than wins (assuming that there are other informed verifiers).

Intuitively, the system is designed to provide (1) liveness via Recommenders who are incentivized to introduce new models to the system as well as (2) safety via Verifiers who ensure that non-competitive (or non-verified) models accumulate in the system.

4 Building Utilities

The system also requires costly signaling in accordance with principles of game theory for signaling games as well as financial engineering. For this purpose, we assume existence of a cryptographic security token system that distributes tokens to Recommenders and Verifiers in exchange of financial investments.

These tokens are used in the dynamics of the game. In addition, there may be rent to be collected by allowing other agents (senders and receivers) to use the models in the ecosystem for other applications, where AI could be used productively. The rent for the models can be calculated by classical financial engineering approaches (e.g., CAPM, Capital Asset Pricing Models).

5 An Example System for FinTech

For the sake of concreteness, we use an example from FinTech, though the similar structures can be used for other domains *mutatis mutandis*. Modern FinTech builds on a wide range of disciplines: namely, from computer and data sciences, at one end of the spectrum, to finance, artificial intelligence, game theory and mathematics at the other. It is focused on a central aspect of modern economic life: namely, finance, a multi-faceted subject that draws on ideas from mathematics, economics, psychology and other social and political sciences, and increasingly, information technology and how it connects to society and social networks. In the last half of the century, with expanding data sources and increasing computing speed, finance has become more reliant on statistics, data and econometric models, slowly paving the way to newer forms of financial engineering and technologies as well as their management and regulation – often by computational means.

"FinTech" refers to financial sector innovations involving technology-enabled economic models that can eliminate market inefficiency, support synchronicity in the market and improve risks and liquidity, facilitate disintermediation, revolutionize how existing and emerging firms create and deliver products and services, address privacy, trust and personalization, regulatory and law-enforcement challenges, and seed opportunities for inclusive growth. Some of these innovations could substantially improve the economic life of a larger number of citizens, but would also require the development of new approaches to understand their opportunities and challenges.

Nonetheless, the evolving applications of FinTech has encountered a methodological problem, common to many domains using Data Science and Artificial Intelligence. Namely, (a) How does one quantitatively measure how much better an AI-based FinTech system performs in comparison to traditional approaches from statistical inference, econometrics, model-based (Bayesian) analysis, etc.; (b) How does one disentangle the improvements attributable to model selection, data size, special purpose computation (e.g., GPU or TPU), etc.? (c) How does one decide how to design future systems for a suitable application (e.g., ones with information asymmetry and illiquidity), a suitable computational infrastructure (e.g., clouds with special purpose software like Map-reduce or BigQuery) and a suitable data sets (e.g., social media data vs. cancer genomics data)?

A general goal is to carry out an empirical analysis with a prototype one may plan to build (more details are available in the full paper). This prototype will involve maintaining an ecosystem of models, with additional information such as how it was introduced, what information was given, a preliminary empirical

Bayesian evaluation of its goodness (e.g., rank), competition involving additional verifiers and the results. For succinctness, it may display an aggregated rank for each model in the ecosystem (specific to a particular domain).

The prototype could thus be used for evaluating Data Science, Machine Learning and AI based FinTech systems, to be used by the applied finance community. This prototype will be incorporating the essence of AI systems to detect statistical arbitrage, pricing models with metrics for *risks* and *liquidity* and the changes in the underlying market and regulatory microstructures[1]. This tool could be useful in gathering real time information about various FinTech and RegTech technologies in an international setting (US, India and China) and setting up the technology evaluation on a broader dataset. Its users may use this prototype environment to set up a base for economic, financial, mathematical and computational scientists to work together and solve complex fundamental problems in economics, computing, intelligence and learning.

5.1 Component 1

Rationale: Currently most powerful AI approaches are based on supervised learning. They are fairly simple in the formulation of the problems, but have performed surprisingly well in tasks that are mostly attributed to complex human skills: handwriting recognition, spam detection, image recognition, tagging humans in images, creating captions for images, language translation, etc., to name a few. It has been argued that such approaches, be they as successful as they may, only capture roughly one second of human cognition – roughly the tasks that can be performed by a large group of Mechanical Turks, engaged in labeling raw data.

Formally speaking, in the classical context, AI (more precisely, Machine Learning) deals with two fundamental spaces: The first space $\mathcal{D}$ consists of data points (e.g., point clouds in a high dimensional space) and the second space $\mathcal{M}$ consists of learning models (e.g., parameters of a distributions or weights of a multi-layer artificial neural net, etc.) In statistical learning, $\mathcal{M}$ is usually a space of statistical models $\{p(x, M) : M \in \mathcal{M}\}$ in the *generative* case or $\{p(y|x; M) : M \in \mathcal{M}\}$ in the *discriminative* case. The space $\mathcal{M}$ can be either a low dimensional parametric space or the space of all suitably sparse models in non-parametric space. While classical statistical estimation theory has focused on the former, there is a significant emphases on the later in machine learning – our primary focus here.

A machine learning algorithm selects a model $M \in \mathcal{M}$ based on a training sample $\{(x_i, y_i)_{i=1}^n \subseteq \mathcal{X} \times \mathcal{Y}\}$. Usually the selection of the model is formalized as an optimization problem in two parts: (i) guarding against underfitting:

[1] We will use AI to broadly refer to many diverse approaches that include Statistical Inference, Econometrics, Data Science, Big Data, Probably Approximate Computational (PAC) Learning, Shallow and Deep Neural Nets, Genetic Algorithms, Heuristics, Machine Learning, etc., all of which employ large-scale data and scalable computing.

by maximizing likelihood, margin, distance/divergence or utility (or minimizing a loss function) together with (ii) guarding against overftting: by regularizing with shrinkage, entropy, information or sparsity (or a proxy such as L_1 norms), etc. There is generally a lack of an all-encompassing theory to compare various model selection approaches used by machine learning software (even in a specific domain such as FinTech) and nonconclusive anecdotal arguments based on empirical studies on benchmark data sets have been poor substitute for a deeper understanding.

Approach: The prototype allows one to study a wide class of AI algorithms developed specifically for FinTech. For this purposes, we may formalize our approach in the language of *multiple hypothesis testing*: namely, each model $m_i \in \mathcal{M}$ learned from a training data set $D_T = \{(x_i, y_i)_{i=1}^n \subseteq \mathcal{X} \times \mathcal{Y}$ corresponds to a hypothesis that it will perform with a specific "score" s_i on an unseen cross validating data set $D_V = \{(x'_i, y'_i)_{i=1}^m \subseteq \mathcal{X} \times \mathcal{Y}$. In particular, the prototype may use Efron's Empirical Bayes [1–3] approaches to control false discovery rates (fdr) and measure how well models from each family of machine learning algorithms is likely to win such a "horse race." This likelihood could be used to determine if a recommended model would be eligible for "competition," and thus included in the eco-system.

For our purposes, we may consider methods available in various AI open source platforms: e.g., WEKA, H2O, TensorFlow, OpenAI, Caffe, CNTK, Mahout, etc. We may deploy the prototype in a real setting, which is likely to closely follow various developments in AI that could be applied to finance data, and may implement most of the commonly used models (e.g., regressions, classification trees, neural nets (DNN and ENN: respectively Deep and Evolutionary Neural Nets), etc.). The data source the prototype may use would be derived from proprietary financial data collected by a large bank. The multiple hypotheses testing techniques, outlined earlier, would be applied to models derived from training data spanning several years with the cross validation applied to the last year of data. The overall success of the entire framework will be tested by applying only the truly successful machine learning models to financial data arriving in real-time over six months subsequent to the end of full analysis.

This prototype will support an ecosystem of models that have already been tried. When a new model is recommended this approach may be used to perform a preliminary analysis of the model being introduced and a competition is deemed eligible if and only if the proposed model passes the analysis outlined here.

5.2 Component 2

Rationale: Practically all machine learning algorithms currently in use suffer from several shortcomings that make them less than ideal for FinTech applications: (i) these algorithms assume a stationary distributions over $\mathcal{X} \times \mathcal{Y}$, and hence only capture an instantaneous response to the new incoming data; (ii) they are "black boxes," and are difficult to interpret or use in a strategic interventions [4,5]; and (iii) they are blind to "black swan events," costly adversarial events that are rare but plausible. In order to remedy these disadvantages, machine learning algorithms must understand the causal structures in the data and be amenable to stress testing that require causal generative models consistent with causal structures.

Approach: In order to address these issues, one may explore construction of graphical models that capture causal structures via a DAG (directed acyclic graphs), whose directed edges correspond to Suppes' notions of prima-facie causes. These models (SBCN: Suppes-Bayes Causal Nets) are regularized using BIC (Bayes Information Criterion) to eliminate spurious prima-facie causes and only retain genuine causes, supported by the training data.

Suppes notion of *probabilistic causation* is based on two ideas: (a) temporal priority (causes precede effect) and (b) probability raising (causes raise the conditional probability of the effect in the presence of the causes relative to its absence); these two notions are easily captured in a logic, called PCTL (probabilistic computational tree logic) [6,7], which supports efficient polynomial time model checking algorithms.

Once an SBCN is constructed over financial factors (e.g., Fama French Five Factor Models), it is possible to traverse the graph to generate plausible adversarial rare trajectories that stress test a particular discriminative model (as the ones described earlier). Using these stress testing algorithms [8], we plan to analyze the best AI models selected earlier, to further identify robust profit-generating models.

If the recommended model enters a competition, the recommender may publish the results of the causal analysis.

6 Conclusion

These notes only sketch out a game theoretic model for creating AI of the future. These notes currently lack a detailed analysis (both theoretical and empirical) to ensure that the prototype may be competitive to the traditional approach. Also, although its connections to GAN, ABC and Turing Learning are fairly straightforward, they will be relegated to future sequels.

Nonetheless, since the fundamental research questions in the AI (with applications to FinTech) arena have not been well defined, we believe that the proposed research will create an opportunity for thought leadership.

Acknowledgement. We wish to thank Marti Subramanyam, Abhinav Tamskar and Vasant Dhar (all of NYU) for their questions and comments. We also wish to thank founders, members, investors and advisors of the startup Prospero.ai for their interest in creating a prototype to test out some of the core ideas in AI. We also acknowledge NYU for their financial support to carry out the research.

References

1. Efron, B., Hastie, T.: Computer Age Statistical Inference: Algorithms, Evidence and Data Science. Cambridge University Press, New York (2016)
2. Efron, B.: Bayes, oracle bayes, and empirical bayes (2017)
3. Wager, S., Hastie, T., Efron, B.: Confidence intervals for random forests: the jackknife and the infinitesimal jackknife. J. Mach. Learn. Res. **15**(1), 1625–1651 (2014)
4. Goodfellow, I.J., Bengio, Y., Courville, A.C.: Deep Learning. Adaptive Computation and Machine Learning. MIT Press, Cambridge (2016)
5. Goodfellow, I.J.: NIPS 2016 tutorial: generative adversarial networks. CoRR, vol. abs/1701.00160 (2017)
6. Kleinberg, S., Mishra, B.: The temporal logic of token causes. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, 9–13 May 2010 (2010)
7. Kleinberg, S., Mishra, B.: The temporal logic of causal structures,. In: UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, 18–21 June 2009, pp. 303–312 (2009)
8. Gao, G., Mishra, B., Ramazzotti, D.: Efficient simulation of financial stress testing scenarios with suppes-bayes causal networks. In: International Conference on Computational Science, ICCS 2017, Zurich, Switzerland, June 12–14 2017, pp. 272–284 (2017)