Toward A “Standard Model” of Machine Learning

Zhiting Hu
Assistant Professor
Halicioglu Data Science Institute
Computer Science and Engineering
UC San Diego

06/12/2023
The universe of problems ML/AI is trying to solve
Experience of all kinds

Type-2 diabetes is 90% more common than type-1

Data examples  Rules/Constraints  Knowledge graphs  Rewards  Auxiliary agents

Adversaries  Master classes

• And all combinations of such
• Interpolations between such
• ...

Knowledge graphs
Human learning vs machine learning

Data examples

Rules/Constraints

Knowledge graphs

Rewards

Auxiliary agents

- And all combinations of such
- Interpolations between such
- ...

Type-2 diabetes is 90% more common than type-1

Adversaries

Master classes
The zoo of ML algorithms

- maximum likelihood estimation
- data re-weighting
- inverse RL
- policy optimization
- active learning
- reinforcement learning as inference
- actor-critic
- imitation learning
- reward-augmented maximum likelihood
- softmax policy gradient
- posterior regularization
- constraint-driven learning
- adversarial domain adaptation
- GANs
- generalized expectation
- learning from measurements
- label smoothing
- intrinsic reward
- knowledge distillation
- intrinsic reward
- prediction minimization
- regularized Bayes
- energy-based GANs
- weak/distant supervision
- active learning
- intrinsic reward
- knowledge distillation
- prediction minimization
- regularized Bayes
- energy-based GANs
- weak/distant supervision
Physics in the 1800’s

• Electricity & magnetism:
  • Coulomb’s law, Ampère, Faraday, ...

• Theory of light beams:
  • Particle theory: Isaac Newton, Laplace, Plank
  • Wave theory: Grimaldi, Chris Huygens, Thomas Young, Maxwell

• Law of gravity
  • Aristotle, Galileo, Newton, …
**Standard Model in Physics**

**Maxwell’s Eqns:**

1. **original form**
   - \(\varepsilon_{uvk}\lambda \partial_v F_{k\lambda} = 0\)
   - \(\nabla \cdot \mathbf{D} = \rho_v\)
   - \(\nabla \cdot \mathbf{B} = 0\)
   - \(\nabla \times \mathbf{E} = \frac{\partial \mathbf{B}}{\partial t}\)
   - \(\nabla \times \mathbf{H} = \frac{\partial \mathbf{D}}{\partial t} + \mathbf{J}\)

2. **Simplified w/ rotational symmetry**
   - \(\nabla \cdot \mathbf{D} = \rho_v\)
   - \(\nabla \cdot \mathbf{B} = 0\)

3. **Further simplified w/ symmetry of special relativity**
   - \(\varepsilon_{uvk}\lambda \partial_v F_{k\lambda} = 0\)
   - \(\nabla \cdot \mathbf{D} = \rho_v\)

4. **Standard Model w/ Yang-Mills theory and US(3) symmetry**
   - \(\mathcal{L}_{gf} = -\frac{1}{2} \text{Tr}(F^2)\)
   - \(-\frac{1}{4} F_{\mu\nu} F^{\mu\nu}\)

**Diverse electromagnetic theories**

- **Maxwell’s Eqns:**
  - **original form**
  - **simplified w/ rotational symmetry**
  - **further simplified w/ symmetry of special relativity**
  - **Standard Model w/ Yang-Mills theory and US(3) symmetry**

**Unification of fundamental forces?**
Quest for more standardized, unified ML principles

EDITORIAL
Toward a Unified Science of Machine Learning
(P. Langley, 1989)

A Unifying Review of Linear Gaussian Models

Sam Roweis*
Computation and Neural Systems, California Institute of Technology, Pasadena, CA 91125, U.S.A.

Zoubin Ghahramani*
Department of Computer Science, University of Toronto, Toronto, Canada

*Corresponding author.
Quest for more standardized, unified ML principles

Is Large Language Model (LLM) the answer?

“Self-supervised” learning + large (text) data

😍 Limited understanding of the world

John put a book on the desk.

…

Mary took the book. She placed it on the sofa.

…

Where was the book?

ChatGPT

It was on the desk. ❌

figure credit: Voicebot.ai
Quest for more standardized, unified ML principles
Is Large Language Model (LLM) the answer?

“Self-supervised” learning + large (text) data

🤔 Limited understanding of the world

John put a book on the desk.

Still need more types of experience through richer learning mechanisms

It was on the desk. X

ChatGPT
A “Standard Model” of Machine Learning

Different Model Types
- Graphical Models
- Deep Neural Network
- Symbolic Knowledge
- World Model

Experience $f$
- Data
- Knowledge
- Other Models
- Reward

Uncertainty (Self-regularization)
- World Model

Standard Equation (SE)

Hu and Xing, Towards A ‘Standard Model’ of Machine Learning, Harvard Data Science Review, 2022
A “Standard Model” of Machine Learning

\[
\min_{q, \theta} - \mathbb{E}_{q(x, y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_\theta(x, y) \right) - \beta \mathbb{H}(q)
\]

3 terms:
- **Experience** (exogenous regularizations)
  - e.g. data examples, reward
- **Divergence** (fitness)
  - e.g. Cross Entropy
- **Uncertainty** (self-regularization)
  - e.g. Shannon entropy

Textbook: \( f(x, y) \)

Teacher: \( q(x, y) \)

Student: \( p_\theta(x, y) \)

Uncertainty

Hu and Xing, *Towards A ‘Standard Model’ of Machine Learning*, Harvard Data Science Review, 2022
A “Standard Model” of Machine Learning

\[
\min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_{\theta}(x, y) \right) - \beta \mathbb{H}(q)
\]

Hu and Xing, *Towards A ‘Standard Model’ of Machine Learning*, Harvard Data Science Review, 2022
“Standard Model” encompasses well-known ML algorithms as special cases

\[
\min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_\theta (x, y) \right) - \beta \mathbb{H}(q)
\]
"Standard Model" encompasses well-known ML algorithms as special cases

\[
\min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathcal{D} \left( q(x, y), p_{\theta}(x, y) \right) - \beta \mathcal{H}(q)
\]

| Experience type | Experience function \( f \) | Divergence \( \mathcal{D} \) | \( \alpha \) | \( \beta \) | Algorithm |
|-----------------|-------------------------------|-------------------|--------|--------|-----------|
| Data instances  |                               |                   |        |        |           |
| \( f_{\text{data}}(x; D) \) | CE                           | 1                 | 1      |        | Unsupervised MLE |
| \( f_{\text{data}}(x, y; D) \) | CE                           | 1                 | \( \epsilon \) |        | Supervised MLE |
| \( f_{\text{data-self}}(x, y; D) \) | CE                           | 1                 | \( \epsilon \) |        | Self-supervised MLE |
| \( f_{\text{data-w}}(t; D) \) | CE                           | 1                 | \( \epsilon \) |        | Data Re-weighting |
| \( f_{\text{data-aug}}(t; D) \) | CE                           | 1                 | \( \epsilon \) |        | Data Augmentation |
| \( f_{\text{active}}(x, y; D) \) | CE                           | 1                 | \( \epsilon \) |        | Active Learning (Ertekin et al., 2007) |
"Standard Model" encompasses well-known ML algorithms as special cases

\[
\min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_\theta(x, y) \right) - \beta \mathbb{H}(q)
\]

| Experience type   | Experience function $f$ | Divergence $\mathbb{D}$ | $\alpha$ | $\beta$ | Algorithm                      |
|-------------------|-------------------------|--------------------------|----------|---------|-------------------------------|
| Data instances    |                         |                          |          |         |                               |
| $f_{\text{data}}(x; D)$ | CE                      | 1                        | 1        |         | Unsupervised MLE              |
| $f_{\text{data}}(x, y; D)$ | CE                      | 1                        | $\epsilon$ |         | Supervised MLE                |
| $f_{\text{data-self}}(x, y; D)$ | CE                      | 1                        | $\epsilon$ |         | Self-supervised MLE           |
| $f_{\text{data-w}}(t; D)$ | CE                      | 1                        | $\epsilon$ |         | Data Re-weighting             |
| $f_{\text{data-aug}}(t; D)$ | CE                      | 1                        | $\epsilon$ |         | Data Augmentation              |
| $f_{\text{active}}(x, y; D)$ | CE                      | 1                        | $\epsilon$ |         | Active Learning (Ertekin et al., 2007) |

$$f_{\text{data}}(x, y; D) := \log \mathbb{E}_{(x^*, y^*) \sim D} \left[ \mathbb{1}_{(x^*, y^*)}(x, y) \right]$$

$$q(x, y) = \tilde{p}_{\text{data}}(x, y)$$

$$\min_\theta - \mathbb{E}_q \left[ \log p_\theta(x, y) \right]$$

(Negative data log-likelihood)
“Standard Model” encompasses well-known ML algorithms as special cases

\[
\min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_\theta(x, y) \right) - \beta \mathbb{H}(q)
\]

| Experience type | Experience function \( f \) | Divergence \( \mathbb{D} \) | \( \alpha \) | \( \beta \) | Algorithm |
|-----------------|-------------------------------|----------------|-----|-----|----------|
| Reward          | \( \log Q^\theta(x, y) \)    | CE             | 1   | 1   | Policy Gradient |
|                 | \( \log Q^\theta(x, y) + Q^{in, \theta}(x, y) \) | CE             | 1   | 1   | + Intrinsic Reward |
|                 | \( Q^\theta(x, y) \)         | CE             | \( \rho > 0 \) | \( \rho > 0 \) | RL as Inference |
“Standard Model” encompasses well-known ML algorithms as special cases

\[
\min_{\theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_\theta(x, y) \right) - \beta H(q)
\]

| Experience type | Experience function \( f \) | Divergence \( \mathbb{D} \) | \( \alpha \) | \( \beta \) | Algorithm |
|-----------------|-------------------------------|------------------|---------|---------|----------|
| Knowledge       | \( f_{\text{rule}}(x, y) \)   | CE               | 1       | 1       | Posterior Regularization (Ganchev et al., 2010) |
|                 | \( f_{\text{rule}}(x, y) \)   | CE               | \( \mathbb{R} \) | 1       | Unified EM (Samdani et al., 2012) |
“Standard Model” encompasses well-known ML algorithms as special cases

\[
\min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_{\theta}(x, y) \right) - \beta \mathbb{H}(q)
\]

| Experience type | Experience function \( f \) | Divergence \( \mathbb{D} \) | \( \alpha \) | \( \beta \) | Algorithm |
|-----------------|-------------------------------|-----------------|-------|------|------------|
| Model           | \( f_{\text{mimicking}}(x, y; D) \) | CE              | 1     | \( \epsilon \) | Knowledge Distillation (G. Hinton et al., 2015) |
"Standard Model" encompasses well-known ML algorithms as special cases

\[
\min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_\theta(x, y) \right) - \beta \mathbb{H}(q)
\]

| Experience type          | Experience function f | Divergence \(\mathbb{D}\) | \(\alpha\) | \(\beta\) | Algorithm                        |
|--------------------------|-----------------------|-----------------------------|-----------|-----------|----------------------------------|
| binary classifier        |                        | JSD                         | 0         | 1         | Vanilla GAN (Goodfellow et al., 2014) |
| discriminator            | \(f\)-divergence      |                             | 0         | 1         | f-GAN (Nowozin et al., 2016)        |
| 1-Lipschitz discriminator | \(W_1\) distance      |                             | 0         | 1         | WGAN (Arjovsky et al., 2017)        |
| 1-Lipschitz discriminator | KL                    |                             | 0         | 1         | PPO-GAN (Y. Wu et al., 2020)        |
”Standard Model” encompasses well-known ML algorithms as special cases

\[
\min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathcal{D} \left( q(x, y), p_\theta(x, y) \right) - \beta \mathbb{H}(q)
\]

| Experience type | Experience function \( f \) | Divergence \( \mathcal{D} \) | \( \alpha \) | \( \beta \) | Algorithm |
|-----------------|-----------------------------|-----------------|-------|-------|-----------|
| Online          | \( f_\tau(t) \)             | CE              | \( \rho > 0 \) | \( \rho > 0 \) | Multiplicative Weights (Freund & Schapire, 1997) |
“Standard Model” encompasses well-known ML algorithms as special cases

| Experience type | Experience function $f$ | Divergence $\mathbb{D}$ | $\alpha$ | $\beta$ | Algorithm                                      |
|-----------------|-------------------------|--------------------------|---------|---------|-----------------------------------------------|
| Data instances  | $f_{\text{data}}(x; \mathcal{D})$ | CE                        | 1       | 1       | Unsupervised MLE                              |
|                 | $f_{\text{data}}(x, y; \mathcal{D})$ | CE                        | 1       | $\epsilon$ | Supervised MLE                               |
|                 | $f_{\text{data-self}}(x, y; \mathcal{D})$ | CE                        | 1       | $\epsilon$ | Self-supervised MLE                           |
|                 | $f_{\text{data-w}}(t; \mathcal{D})$ | CE                        | 1       | $\epsilon$ | Data Re-weighting                             |
|                 | $f_{\text{data-aug}}(t; \mathcal{D})$ | CE                        | 1       | $\epsilon$ | Data Augmentation                             |
|                 | $f_{\text{active}}(x, y; \mathcal{D})$ | CE                        | 1       | $\epsilon$ | Active Learning (Ertekin et al., 2007)        |
| Knowledge       | $f_{\text{rule}}(x, y)$ | CE                        | 1       | 1       | Posterior Regularization (Ganchev et al., 2010) |
|                 | $f_{\text{rule}}(x, y)$ | $\mathbb{R}$          | $\epsilon$ | 1       | Unified EM (Samdani et al., 2012)             |
| Reward          | $\log Q^\theta(x, y)$       | CE                        | 1       | 1       | Policy Gradient                              |
|                 | $\log Q^\theta(x, y) + Q^{\text{in}, \theta}(x, y)$ | CE                        | 1       | 1       | + Intrinsic Reward                           |
|                 | $Q^\theta(x, y)$             | CE                        | $\rho > 0$ | $\rho > 0$ | RL as Inference                               |
| Model           | $f_{\text{mimicking}}(x, y; \mathcal{D})$ | CE                        | 1       | $\epsilon$ | Knowledge Distillation (G. Hinton et al., 2015) |
| Variational     | binary classifier | JSD                      | 0       | 1       | Vanilla GAN (Goodfellow et al., 2014)         |
|                 | discriminator            | $f$-divergence            | 0       | 1       | f-GAN (Nowozin et al., 2016)                  |
|                 | 1-Lipschitz discriminator | $W_1$ distance          | 0       | 1       | WGAN (Arjovsky et al., 2017)                  |
|                 | 1-Lipschitz discriminator | KL                       | 0       | 1       | PPO-GAN (Y. Wu et al., 2020)                  |
| Online          | $f_r(t)$                  | CE                        | $\rho > 0$ | $\rho > 0$ | Multiplicative Weights (Freund & Schapire, 1997) |

Table 1. Example configurations of the components in the standard equation (Eqs. 3.1, 3.2), which recover different existing algorithms. Here, ‘CE’ means Cross Entropy; ‘JSD’ is the Jensen-Shannon divergence; ‘W$_1$ dist.’ is the first-order Wasserstein distance; and ‘KL’ is the KL divergence. Refer to Sections 4, 5, and 6 for more details.
Applications: “Panoramic” learning with ALL experience

All available experience

Arbitrary model

Data examples

Rules/Constraints

Knowledge graphs

Rewards

Auxiliary agents

Type-2 diabetes is 90% more common than type-1

Adversaries

Master classes

Applications: “Panoramic” learning with ALL experience

…

- And all combinations of such
- Interpolations between such
- …
App (1): Using *symbolic knowledge* to learn *neural networks*

\[
\min_{q, \theta} - \alpha \mathbb{H}(q) + \beta \mathbb{D} \left( q(x, y), p_\theta(x, y) \right) - \mathbb{E}_{q(x, y)} \left[ f(x, y) \right]
\]

Hu et al., 2016, “Harnessing Deep Neural Networks with Logic Rules”
Hu et al., 2020, “Deep Generative Models with Learnable Knowledge Constraints”
Tan et al., 2020, “Summarizing Text on Any Aspects: A Knowledge-Informed Weakly-Supervised Approach”
App (2): Using *neural networks* to "learn" *symbolic knowledge*

\[
\min_{q, \theta} - \alpha H(q) + \beta D(q(x, y), p_\theta(x, y)) - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right]
\]

- \(\theta\): graph structure to be learned
- \(p_\theta\): a simulation model generating medical task samples \((x, y)\) based on the knowledge graph \(\theta\)

Measuring likelihood of sample \((x, y)\) under a trained *medical neural model*

---

Hao, Tan et al., 2022, “BertNet: Harvesting Knowledge Graphs from Pretrained Language Models”
App (2): Using **neural networks** to “learn” **symbolic knowledge**

\[
\min_{q, \theta} - \alpha \mathcal{H}(q) + \beta \mathbb{D}(q(x, y), p_\theta(x, y)) - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right]
\]

| Head entity | Relation   | Tail entity | Head entity          | Relation                      | Tail entity              |
|-------------|------------|-------------|----------------------|------------------------------|--------------------------|
| exercise    | prevent    | obesity     | students             | worth celebrating            | graduate                 |
| apple       | business   | Mac         | newborn              | can but not good at          | sit                      |
| sleep       | prevent    | illness     | social worker        | can help                     | foster child             |
| mall        | place for  | shopping    | honey                | ingredient for               | honey cake               |
| gym         | place for  | sweat       | cabbage              | ingredient for               | cabbage salad            |
| wheat       | source of  | flour       | China                | separated by the ocean       | Japan                    |
| oil         | source of  | fuel        | Africa               | separated by the ocean       | Europe                   |

Figure 4: Examples of knowledge tuples harvested from ROBERTA-LARGE with MULTI-PROMPTS.

Hao, Tan et al., 2022, “BertNet: Harvesting Knowledge Graphs from Pretrained Language Models”
App (3): Building **World Models** beyond **Language Models**

\[
\min_{q, \theta} - \alpha \mathbb{H}(q) + \beta \mathbb{D}(q(x, y), p_\theta(x, y)) - \mathbb{E}_{q(x,y)}[f(x, y)]
\]

- **Language models**

---

Xiang, Tao et al., 2023, “Language Models Meet World Models: Embodied Experiences Enhance Language Models”
App (3): Building **World Models** beyond **Language Models**

\[
\min_{q, \theta} - \alpha \mathbb{H}(q) + \beta \mathbb{D} \left( q(x, y), p_{\theta}(x, y) \right) - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right]
\]

John put a *book* on the *desk*.

...  
Mary took the *book*.  
She placed it on the *sofa*.  
...  
Where was the *book*?

ChatGPT: It was on the *desk*.  
WM (small-size): It was on the *sofa*.

Xiang, Tao et al., 2023, “Language Models Meet World Models: Embodied Experiences Enhance Language Models”
App (3): Building **World Models** beyond **Language Models**

\[
\min_{q, \theta} \left( -\alpha \mathbb{H}(q) + \beta \mathcal{D}(q(x, y), p_\theta(x, y)) - \mathbb{E}_{q(x, y)} \left[ f(x, y) \right] \right)
\]

Xiang, Tao et al., 2023, “Language Models Meet World Models: Embodied Experiences Enhance Language Models”
Summary

- A “Standard Model” of machine learning
  \[ \min_{q, \theta} - \mathbb{E}_{q(x,y)} \left[ f(x, y) \right] + \alpha \mathbb{D} \left( q(x, y), p_{\theta}(x, y) \right) - \beta \mathbb{H}(q) \]

- “Panoramic learning” with ALL experience
  - Neuro-symbolic learning
  - Building world models

---

**Figure 4:** Examples of knowledge tuples harvested from ROBERTA-LARGE with MULTI-PROMPTS.