Deep Learning Applications in the Natural Sciences

P. Baldi

Department of Computer Science
Institute for Genomics and Bioinformatics
Center for Machine Learning and Intelligent Systems
University of California, Irvine
Scientific Discovery Drivers

1. Data (Sensors, Instruments, Data Bases, Internet, Storage…)

2. Computing (Clusters, Cloud, GPUs…)

3. Machine Learning (AI, Statistics, Data Mining, Algorithms…)
Scientific Discovery
Cosmological frontier
NEW HiSeq 2500

Genetic frontier
Scientific Discovery Drivers

1. Data (Sensors, Instruments, Data Bases, Internet, Storage…)
   - Exponential Growth
   - Uneveness across Fields

2. Computing (Clusters, Cloud, GPUs…)

![Data Center Image](image_url)
Scientific Discovery Drivers

1. Data (Sensors, Instruments, Data Bases, Internet, Storage…)

2. Computing (Clusters, Cloud, GPUs…)

3. Machine Learning (AI, Statistics, Data Mining, Algorithms…)
Deep Learning

Training set: \((x_i, y_i)\)
or \((x_i, x_i)\)
i = 1, \ldots, m

\(n_1\)

\(n_j\)

\(n_0\)
\[ a = \sum_{i=1}^{N} W_i x_i + \text{Bias} \]

output = \text{Threshold}(a)

where \text{Threshold}(a) = \begin{cases} 
1, & \text{for all } a \leq 0 \\
1, & \text{for all } a > 0 
\end{cases}
Computer Vision - Image Classification

- **Imagenet**
- Over 1 million images, 1000 classes, different sizes, avg 482x415, color
- 16.42% Deep CNN dropout in 2012
- 6.66% 22 layer CNN (GoogLeNet) in 2014
- 4.9% (Google, Microsoft) super-human performance in 2015

Sources:
- Benenson, http://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results.html
DL Theory

• 1-Layer Networks
  – Perceptron theorem
  – Linear regression; Logistic regression;
  – Statistical theory and design (top layer)

• 1.5-Layer Networks
  – Bottom layer = random    (Johnson-Lindenstrauss)
  – Bottom layer = similarities (dot products or kernels) → SVM
DL Theory

• 2-Layers Networks
  – Universal approximation
  – Autoencoders (compressive, expansive)
  – Linear autoencoders (PCA and landscape)
  – Non-linear autoencoders (Boolean autoencoder, clustering, NP-completeness)
DL Theory

• **L-Layer Networks**
  – Linear
  – Boolean unrestricted
  – Local learning and its limitations (generalization of Hebb)
  – Optimality of Backpropagation
  – Design (Weight sharing, Compression and Dark Knowledge, etc)
  – Dropout, Initialization, Learning rates, hyperparameter optimization

• **Recurrent Networks**
  – Hopfield Model and Boltzmann machines
  – Design (DAGs, LSTMs, etc)
DL Theory

• Importance of Group Theory
  – Learning permutations
  – Permutations of the units
  – Symmetries of learning rules
  – Invariant recognition (Lie Groups)

• The Black-Box Problem…..
Two Kinds of Problems and Architectures

1. Input: vectors of fixed size
   (e.g. images in computer vision).

2. Input: structured objects of variable size
   (e.g. sequences in NLP or bioinformatics, graphs in chemistry).
Design (Recursive Architectures)
Graphical Models: DAG-HMM
Recursive Neural Networks: DAG-RNN
Deep Learning in the Natural Sciences

• Physics
  - HEP: Identification of Exotic Particles (<1Å)
  - [Cosmology: Identification of Quasars (10^{26}m)]

• Chemistry
  - Prediction of Molecular Properties and Chemical Reactions (~1-10^2Å)

• Biology
  - Prediction of Protein Structures and Structural Features (10^2-10^4Å)

• Many more
Deep Learning in the Natural Sciences

• Physics
  -HEP: Identification of Exotic Particles (<1Å)

• Chemistry
  -Prediction of Molecular Properties and Chemical Reactions (~1-102Å)

• Biology
  -Prediction of Protein Structures and Structural Features (10^2-10^4Å)
An important, complex, multi-faceted, somewhat ill defined problem.
Deep Learning in Biology
Deep Learning in Biology: Mining Omic Data
Deep Learning in Biology: Mining Omic Data
Progress in Accuracy

• 1978: Chou and Fasman  ~60% (statistical rules)
• 1988: Qian and Sejnowski  ~64% (NNs)
• 1994: Rost and Sander  ~74% (NNs)
• 1999: B. et al.  ~78% (1DBRNNs)
• ...... 
• 2014: Magnan and B.  ~95% (1DBRNNs + homology)
Deep Learning in Biology: Mining Omic Data

Deep Learning in Biology: Mining Omic Data

(In 3D, 8 hidden cubes, etc......)
2D RNNs

\[
\begin{align*}
O_{ij} &= N_O(I_{i,j}, H_{i,j}^{NW}, H_{i,j}^{NE}, H_{i,j}^{SW}, H_{i,j}^{SE}) \\
H_{i,j}^{NE} &= N_{NE}(I_{i,j}, H_{i-1,j}^{NE}, H_{i,j}^{NE}) \\
H_{i,j}^{NW} &= N_{NW}(I_{i,j}, H_{i+1,j}^{NW}, H_{i,j}^{NW}) \\
H_{i,j}^{SW} &= N_{SW}(I_{i,j}, H_{i+1,j}^{SW}, H_{i,j}^{SW}) \\
H_{i,j}^{SE} &= N_{SE}(I_{i,j}, H_{i-1,j}^{SE}, H_{i,j}^{SE})
\end{align*}
\]

Deep Learning

P. Di Lena, K. Nagata, and P. Baldi.
Deep Architectures for Protein Contact Map Prediction.
Bioinformatics, 28, 2449-2457.
Deep Learning in the Natural Sciences

• Physics
  - HEP: Identification of Exotic Particles (<1Å)

• Chemistry
  - Prediction of Molecular Properties and Chemical Reactions (~1-10^2Å)

• Biology
  - Prediction of Protein Structures and Structural Features (10^2-10^4Å)
Prediction of Molecular Properties (Physical, Chemical, Biological)

Melting Temperature? Soluble? Toxic? etc
Data Representations

\[
\text{NC(CO)C(=O)O} \quad 0010001001010001
\]
Data Representations

Problem: molecular graphs are undirected

NC(CO)C(=O)O 0010001001010001
Deep Learning in Chemistry: Predicting Chemical Reactions

\[ \text{RCH}=\text{CH}_2 + \text{HBr} \rightarrow \text{RCH(Br)}=\text{CH}_3 \]

- Many important applications (synthesis, retrosynthesis, etc)
- Three different approaches:
  1. QM
  2. Write a system of rules
  3. Learn the rules from big data
Writing a System of Rules: Reaction Explorer

Table 1. SMIRKS Transformation Rules Corresponding to a Simple Alkene Hydrobromination Reaction Model

<table>
<thead>
<tr>
<th>SMIRKS</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[C:1]=[C:2],[H:3][Cl,Br,I,S(OS=O):4]</td>
<td>alkene, protic acid addition</td>
</tr>
<tr>
<td>[H:3][C:1][C:+:2],[-:4]</td>
<td></td>
</tr>
<tr>
<td>[C:+:1],[-:2]</td>
<td>[C+0:1][+0:2]</td>
</tr>
</tbody>
</table>


- ReactionExplorer System has about 1800 rules
- Covers undergraduate organic chemistry curriculum
- Interactive educational system
- Licensed by Wiley and distributed world-wide
Problems

- Very tedious
- Non-scalable
- Undergraduate chemistry
Deep Learning Chemical Reactions

$$RCH=CH_2 + HBr \rightarrow RCH(Br)-CH_3$$

(a) 

(b)
Siamese Architecture

source                  sink           source                 sink
Deep Learning Chemical Reactions

Figure 2. Overall reaction prediction framework: (a) A user inputs the reactants and conditions. (b) We identify potential electron donors and acceptors using coarse approximations of electron-filled and unfilled MOs. (c) Highly sensitive reactive site classifiers are trained and used to filter out the vast majority of unreactive sites, pruning the space of potential reactions. (d) Reactions are enumerated by pairing filled and unfilled MOs. (e) A ranking model is trained and used to order the reactions, where the best ranking one or few represent the major products. The top-ranked product can be recursively chained to a new instance of the framework for multistep reaction prediction.


Deep Learning in HEP
Standard Model Interactions
(Forces Mediated by Gauge Bosons)

X is any fermion in the Standard Model.

X is electrically charged.

X is any quark.

U is a up-type quark; D is a down-type quark.

L is a lepton and \( \nu \) is the corresponding neutrino.

X is a photon or Z-boson.

X and Y are any two electroweak bosons such that charge is conserved.
Deep Learning in HEP

• Higgs Boson Detection (Nature Communications, 2014)
• Supersymmetry (Nature Communications, 2014)
• Higgs Decay (NIPS 2015, Phys. Rev. Let. 2015)
• Dark Knowledge, DarkMatter (JMLR C&P 2015)
  • Jet substructure, jet tagging, parameterized classifiers

• Common Features and Results:
  - dozens of features: raw + human-derived
  - millions of examples
  - classification problems
  - deep learning outperforms current methods, with or without human-derived features
  - dark knowledge improves shallow architectures
Higgs Boson Detection

Simulation tools:
- MadGraph (collisions)
- PYTHIA (showering and hadronization)
- DELPHES (detector response)

11 M examples
Supervised learning problem:

- Two classes
- 11 million training examples (roughly balanced)
- 28 features
  - 21 low-level features (momenta of particles)
  - 7 high-level features derived by physicists

Data available at archive.ics.uci.edu/ml/datasets/HIGGS
Higgs Boson Detection

Tuning deep neural network architectures.

<table>
<thead>
<tr>
<th>Hyper parameters</th>
<th>Choices</th>
<th>Best:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depth</td>
<td>2, 3, 4, 5, 6 layers</td>
<td>• 5 hidden layers</td>
</tr>
<tr>
<td>Hidden units per layer</td>
<td>100, 200, 300, 500</td>
<td>• 300 neurons per layer</td>
</tr>
<tr>
<td>Learning rate</td>
<td>0.01, 0.05</td>
<td>• Tanh hidden units, sigmoid output</td>
</tr>
<tr>
<td>Weight decay</td>
<td>0, 0.00001</td>
<td>• No pre-training</td>
</tr>
<tr>
<td>Pre-training</td>
<td>none, autoencoder, multi-task autoencoder</td>
<td>• Stochastic gradient descent</td>
</tr>
<tr>
<td>Input features</td>
<td>low-level, high-level complete set</td>
<td>• Mini batches of 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Exponentially-decreasing learning rate</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Momentum increasing from .5 to .99 over 200 epochs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Weight decay = 0.00001</td>
</tr>
</tbody>
</table>
Higgs Boson Detection

Deep network improves AUC by 8%
Dark Knowledge

1. Train deep architecture using **binary targets** (multi-class classification)

2. For each training example retrieve the **soft targets** from the output of the trained deep architecture.

3. Use the soft targets (which contain the dark knowledge), to train a shallow architecture.
Table 7: Performance of shallow networks trained with dark knowledge.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Benchmark 1</th>
<th>Benchmark 2</th>
<th>Benchmark 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN</td>
<td>0.842</td>
<td>0.8786</td>
<td>0.797</td>
</tr>
<tr>
<td>NN w/ dark knowledge</td>
<td>0.850</td>
<td>0.8788</td>
<td>0.799</td>
</tr>
<tr>
<td>DNN</td>
<td>0.885</td>
<td>0.8790</td>
<td>0.802</td>
</tr>
</tbody>
</table>

Dark Knowledge
Jet Substructure and Jet Tagging

Figure 1: Typical jet images from class 1 (single jet) on the left, and class 2 (two overlapping jets) on the right, after preprocessing.

<table>
<thead>
<tr>
<th>Technique</th>
<th>Signal Efficiency</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDT on derived features</td>
<td>90.4%</td>
<td>90.4%</td>
</tr>
<tr>
<td>Shallow NN (32x32 input)</td>
<td>90.2% (0.01%)</td>
<td>90.2% (0.01%)</td>
</tr>
<tr>
<td>Compressed deep NN (32x32 input)</td>
<td>90.2%</td>
<td>90.3%</td>
</tr>
<tr>
<td>Deep NN (32x32 input)</td>
<td>92.7% (0.03%)</td>
<td>92.0% (0.01%)</td>
</tr>
<tr>
<td>Deep NN (48x48 input)</td>
<td>93.0%</td>
<td>92.1%</td>
</tr>
</tbody>
</table>

Table 3: Signal efficiency at 90% background rejection and AUC for each method. The best shallow neural network and the best deep neural network trained on 32 × 32 pixel images were trained with three different random initializations, and we report the mean and standard deviation for each metric.

Figure 3: ROC curves comparing the performance of a deep convolutional network trained on the images, a shallow network trained on the images, and boosted decision trees trained on the designed features.
Jet Substructure

• **Use medium-level variables**
  – Primary vertex (5 variables)
  – Primary tracks (8 variables)
  – Secondary vertices (5 variables)
  – Secondary tracks (8 variables)

• **Use RNNS, CNNs, etc**
The Black-Box Problem

- It is a problem. But:
The Black-Box Problem

• It is a problem.
  1. It is sometimes overblown:
The Black-Box Problem

• It is a problem.
  1. It is sometimes overblown:
     – You do not fully understand your car.
The Black-Box Problem

• It is a problem.
1. It is sometimes overblown:
   – You do not fully understand your car.
   – You do not fully understand your brain.
The Black-Box Problem

• It is a problem.
  1. It is sometimes overblown:
     – You do not fully understand your car.
     – You do not fully understand your brain.
     – The LHC is a collection of black boxes.
The Black-Box Problem

• It is a problem.
1. It is sometimes overblown:
   – You do not fully understand your car.
   – You do not fully understand your brain.
   – The LHC is a collection of black boxes. However the modularity is very important.
The Black-Box Problem

2. It is possible to open up the box and test the output of each neuron under multiple inputs—but very time consuming. Perhaps NNs to understand NNs will be developed.

3. However some degree of opacity is to be expected and inevitable. NNs are a fundamentally different model of computation, where processing and memory are completely interwined, rather than being separated as in current digital computers. In a neural network, data is not stored at a computer address, but rather shattered in each synaptic weight. This is already the case in linear regression.
The Black-Box Problem

4. What does it mean to be a Black Box? What does it mean to understand? To understand is to compress.
The Black-Box Problem

5. The importance of modularity:

Figure 2. Overall reaction prediction framework: (a) A user inputs the reactants and conditions. (b) We identify potential electron donors and acceptors using coarse approximations of electron-filled and unfilled MOs. (c) Highly sensitive reactive site classifiers are trained and used to filter out the vast majority of unreactive sites, pruning the space of potential reactions. (d) Reactions are enumerated by pairing filled and unfilled MOs. (e) A ranking model is trained and used to order the reactions, where the best ranking one or few represent the major products. The top-ranked product can be recursively chained to a new instance of the framework for multistep reaction prediction.
Conclusions

• DL is important but hardly a new idea.

• Examples of applications of deep learning in the natural sciences (Biology, Chemistry, and Physics).

• Natural sciences offer many other challenges and opportunities (QM, Earth Sciences, Astronomy, etc).

• Not only important scientific and technological applications, but also significant challenges, opportunities, and inspiration for machine learning and AI.

• DL yields state of the art performance in HEP and much more to come.
THANK YOU
National Library of Medicine
Twenty Four Years of Growth:
NCBI Data and User Services

Base Pairs ( Millions )
160
140
120
100
80
60
40
20
0

GenBank Base Pairs
Users (Average)

Genome-Bank at NCBI
Entrez
BLAST
OMIM
Genomes
PubMed Central
Access
Human Genome
Genbank
Genome-Wide
Association Studies
NIH Public Access
PubChem
Genetic Testing
Registry
ClinVar
1000 Genomes
MedGen
PubReader

All NCBI Users/Weekday (Million)
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0

Cost per Genome