Deciphering the Contents of Chemically-Trained Neural Networks into Physical Intuition

Eric Berquist

June 15th, 2017

Overview

Machine learning (ML) is seeing rapid growth in areas relevant to quantum chemistry, but how does it work?

- Topic: Are correct ML predictions in quantum chemistry *right for the right reasons*?
- Gap: We don't know if current approaches (ML architectures) will work more complex molecules or properties.
- Rationale: If a ML model is not right for the right reasons, there cannot be an expectation that it is transferable or extendable in any way.

We need to know if ML models are learning chemistry!
(polynomial elephant analogy?)

Overview

- The objective is to quantify what ML models trained on quantum chemical data are learning.
- The central hypothesis is that models are learning about molecular structure identically to how we apply chemical intuition.

This hypothesis will be tested by

- training neural networks (NNs) to replicate literature results,
- "seeing" what the currently-available models have learned using **relevance propagation**,
- attempt to predict more complex molecular properties than those found in the literature, and
- quantify if learning changes for more complex properties.

Disclaimer

The goal of this work is *not* to produce more accurate or more transferable models. The goal is to understand *how* and *why* models make (in)accurate predictions in terms of what they have learned.

Transferability (TODO where to put this?)

Literature usage:

- No need for reparametrization from system to system
- More to do with the input representation than the molecules it can be applied to
- Limited to organic molecules, train small (9 heavy atoms), test larger (10 heavy atoms)
- Charge and spin: neutral and closed-shell singlet

A better definition in terms of examples:

- Does the same model work for optimized and non-equilibrium (MD) structures?
- Does the model work for charged systems?
- Does the model work for systems with unpaired electrons?

What is machine learning?

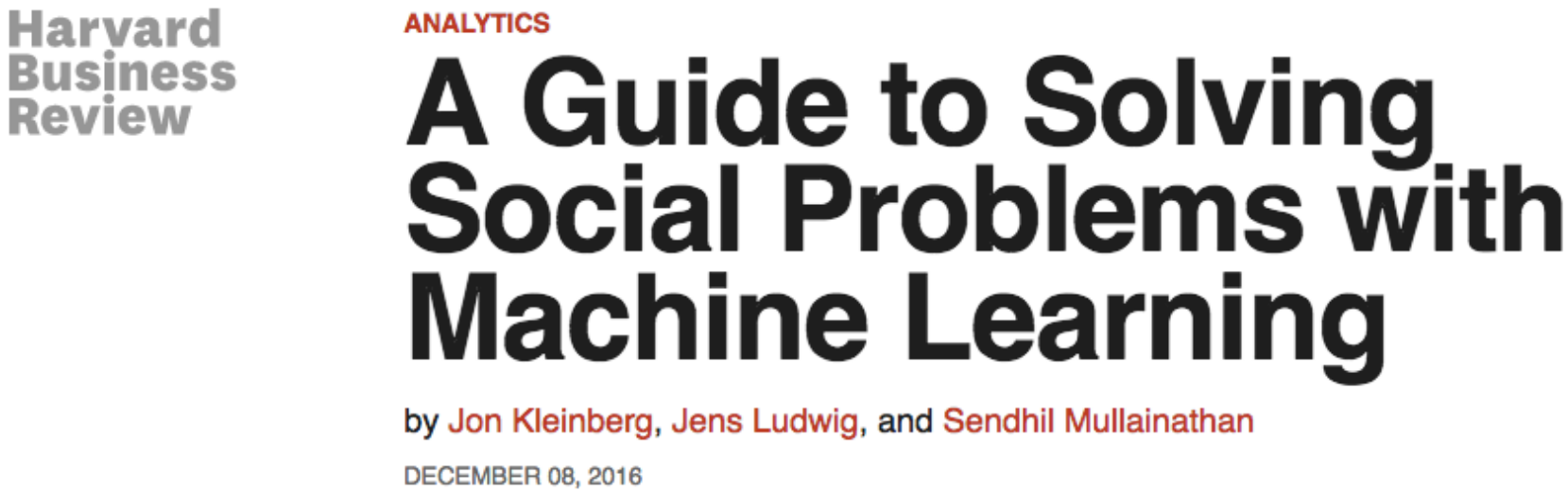
Arthur Samuel, 1959, the subfield of computer science that gives:

> *computers the ability to learn without being explicitly programmed.*

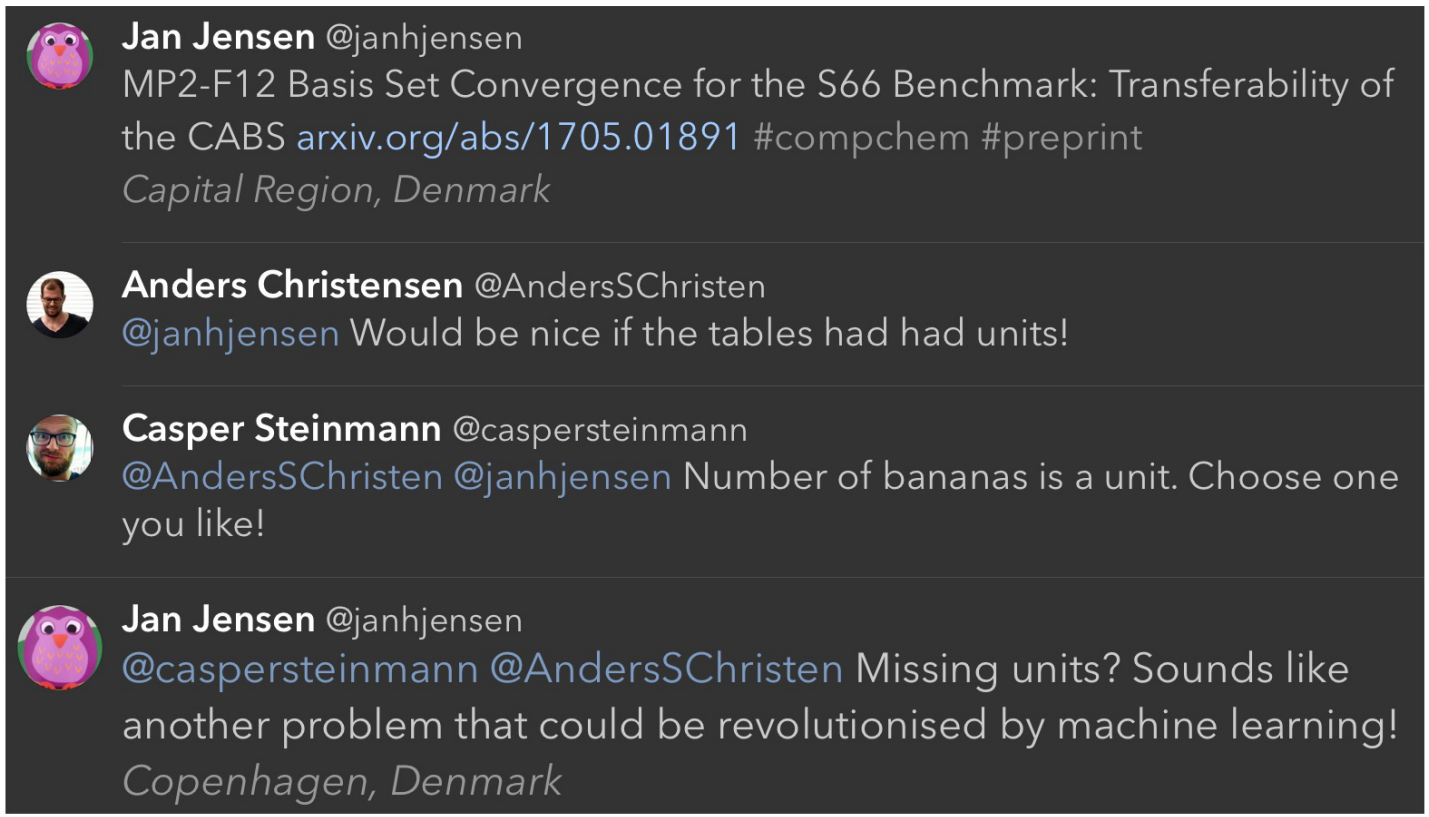
An updated definition: (TODO where is this quote from?)

> *A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.*

Machine learning will solve all our problems



Machine learning will solve all our problems




THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA
ANSWERS

Machine learning has a perception problem

- Machine learning is a “fad” and produces all these great results, but we joke semi-seriously that we don’t know what’s going on under the hood even though it’ll solve all our problems (something about robot overlords).
- introduce black boxes here

Need

- Interpreting the models themselves
- We want the models to learn chemistry!

Rationale

- Building ML models that can do real, useful chemistry in a general manner is impossible without proving meaningfulness (not correctness) of simpler models
- Clearly the dozen or so papers from 2016-2017 show that accurate predictions can be made even under the assumption of black-box models
- Additionally, if we can interpret the model directly, then perhaps eventually we can interpret chemistry using the model itself and not just predictions

Objective

- Peek inside the black box and see if models are "learning chemistry"
- If they aren't, consider other NN architectures (mention ANAKIN-ME, DTNN)
- Is it alright to accept the use of NNs that are not truly transferable (B3LYP, M06)? Maybe this works for prediction results, but we will repeat the history of DFT! (PMWG obituary)

Specific Aims

1. Reproduce existing machine learning models for molecular properties from the literature.
2. Characterize the parameters learned by existing machine learning models from the literature.
3. Train supervised neural networks on complex molecular properties.
4. Characterize the parameters learned for complex molecular properties using unsupervised neural networks.

Background

- Introduction to machine learning
- Simplest form: univariate linear regression
- Neural networks
- LR using a NN
- Pictorial representation of NN structures
- Training a NN
- Relevance propagation: examples
- Relevance propagation: analogies go here

Aim #1: Reproduction of Existing Literature Neural Networks

- Discuss GC architecture, why choose GC architecture over DTNN, ANAKIN-ME, ...?
- Want comparison against literature results (more on this later), these so far are molecular energies only
- Where is the code?
- Why *not* look at molecular energies? Want ML to do spectroscopy too, calculations for which are much more expensive than energies/trajectories.

Aim #2: Characterization of Existing Literature Neural Networks

- Math w/ example pictures
- Expected outcomes

Aim #3: Training Neural Networks for Complex Molecular Properties

- What are the properties, logic behind the choice
- Expected outcomes

Aim #4: Characterization of Novel Neural Networks

- Unsupervised learning (PCA analogy)
- Denoising autoencoder
- Expected outcomes

Approximate Timeline

Specific Aim	Task	# of Months
1	code development: forming pipeline	2
1	model training	1-2
2	code development: adapt LRP to pipeline	2
2	analysis development	1
3	hyperpolarizability calculations	1-2
3	model training	2-3
4	code development: DAE	2
4	model training	2
4	analysis	1
Total		14-17

A future challenge: building databases

- GDB9/QM9 is the most commonly-used training set, the equivalent of the MNIST set of $\tilde{1}$0,000 labeled handwritten digits.
- It is now suffering from the same problem as MNIST: it is too simple and not representative of real-world training cases (molecules).
- Analogy: Pople basis sets (6-31G and derivatives) are still extremely common, not even because we don't know better, but because we "need to compare to past work".
- If a *general and transferable* ML model fails on GDB9, that is a warning sign, but the above cannot be a reason against extending deeper into chemical space for ML model training.

An (imperfect) analogy between neural networks and quantum chemistry

- The fundamental components of the network (kind of neuron activation functions, convolution or direct connection) are like the *Hamiltonian*.
- The number of components in each network layer and the number of layers are like the size of the *basis set*.

A better analogy

between relevance propagation and interaction energy decomposition approaches (SAPT, EDA):

- I am an item

Definitions of trained molecular properties

- Zero-point (vibrational) energy: $E_{\mathrm{ZPVE}} = \frac{1}{2} h \sum_i^{\text{normal modes}} \nu_i$
- Isotropic polarizability (static, $\omega = 0$):
  $\alpha_{\mathrm{iso}} = \bar{\alpha} \equiv \frac{1}{3}(\alpha_{xx} + \alpha_{yy} + \alpha_{zz})$
- Parallel 1st hyperpolarizability (static, $\omega_a = \omega_b = 0$): $\beta_{\parallel} \equiv$

- This is *not* a direct inspection of what the NN has learned!
- Looking directly at NN weights is like looking at MO coefficients. Once the number of them grows, the "importance" of a single one diminishes greatly, and the number of nodes/weights grows even quicker than the number of MO coefficients for a reasonable quantum chemical calculation. The ability for direct inspection becomes impossible.
- Toy models are unlikely to be useful for any kind of understanding the effect of chemical data on NNs because of the complexity of *any* molecule compared to NNs. In a way, a toy or model molecule w/ ab initio calculation can give more insight than a model NN. We are asking NN parameters to be both more efficient and more general than MO coefficients at describing the many-particle wavefunction!

Backup Slides

Hyperpolarizability equations from paper

When the dipole moment coincides with the j-axis, we have

$$\beta_{\parallel} = \frac{3}{5}\beta_j = \frac{1}{5}\sum_{i=x,y,z}(\beta_{iij} + \beta_{iji} + \beta_{jii}), \tag{1}$$

or in the general case,

$$\beta_{\parallel} = \frac{3}{5|\mu|}\sum_{i,j=x,y,z}\beta_{iij}\mu_j, \tag{2}$$

where

$$\beta_{ijk} = \langle\langle \mu_i; \mu_j, \mu_k \rangle\rangle\,. \tag{3}$$