Machine Learning: Theory, Implementation and Practice

Ming-Hen (Henry) Tsai,
a Machine Learning Hacker for Good,
scan33scan33@gmail.com

Jul 8, 2013

Outline

- Introduction
- Machine Learning Algorithms: Discriminative vs Generative Models with Sample Code
- Machine Learning Algorithms: from Theories to Packages
- Machine Learning: Real World Practice
- Review

Outline

What is Machine Learning?

- Applications:

What is Machine Learning?

- Applications:

  search engine, machine translation, spam filtering, medical imaging, etc.

What is Machine Learning?

- Applications:

  search engine, machine translation, spam filtering, medical imaging, etc.
- In one sentence:

What is Machine Learning?

- Applications:

  search engine, machine translation, spam filtering, medical imaging, etc.
- In one sentence:

  learning from *data*

What is Machine Learning?

- Applications:

  search engine, machine translation, spam filtering, medical imaging, etc.
- In one sentence:

  learning from *data*
- How to learn?

What is Machine Learning?

- Applications:

  search engine, machine translation, spam filtering, medical imaging, etc.
- In one sentence:

  learning from *data*
- How to learn?

  machine learning algorithms learn a *model* from *data*

My View for Machine Learning

- Machine Learning is a kind of data summarization technique that human can specify statistical rules in to get results to help people while leaving the judgment of results to the people.

My View for Machine Learning

- Machine Learning is a kind of data summarization technique that human can specify statistical rules in to get results to help people while leaving the judgment of results to the people.
- It is not magic, we have to know something about the task.

My View for Machine Learning

- Machine Learning is a kind of data summarization technique that human can specify statistical rules in to get results to help people while leaving the judgment of results to the people.
- It is not magic, we have to know something about the task.
- It is usually constrained by the reach of human knowledge.

Outline

What are the machine learning algorithms out there?

- Two types: generative model vs discriminative model

What are the machine learning algorithms out there?

- Two types: generative model vs discriminative model
- Generative: models that generate samples for a target task
- Discriminative: models that classify samples for a target task

What are the machine learning algorithms out there?

- Two types: generative model vs discriminative model
- Generative: models that generate samples for a target task
- Discriminative: models that classify samples for a target task
- **End-to-end programs** below: random article generator (generative) and an audio classifier (discriminative).

Random Article Generator – Task Definition

- From some article data, generate a new random article.
- Famous application: SCIgen by MIT.

Random Article Generator – Model

- First order hidden markov chain.
- Maintain a map keyed by word with its value as word-ratio pairs indicating how many times each word is next to the keyed word. (e.g. red $\rightarrow$ [(cat, 2), (dog, 3)])
- Beware of punctuations and special characters.
- Generation rule: given a word, randomly generate the next word weighed by word co-occurence.
- See `https://github.com/scan33scan33/easyml/blob/master/text_generation/article_generator.py` for 36 lines of code.

Audio Classification – Task Definition

- From some audio tracks some labeled positive and some labeled negative, train a model that tells positive ones from negative ones.
- Why? Audio format is something that is not semantically understandable. I want to write some simple programs to make it accessible.

Audio Classification – Model

- Record some audio tracks and do FFT.
- Use Perceptron to train a linear model.
- See `https://github.com/scan33scan33/easyml/blob/master/voice_recognition/voice_auth.py` for a few lines of code.

The Perceptron Algorithm

- Input: n training samples with labels $[(y_1, \mathbf{x}_1), (y_2, \mathbf{x}_2), \ldots, (y_n, \mathbf{x}_n)]$.
- Output: a linear weight vector $\mathbf{w}$.
- Algorithm Framework:
  1. Initialize an initial $\mathbf{w} = [0, \ldots, 0]$.
  2. For each sample, $\mathbf{w} \leftarrow \mathbf{w} - (y_i - \text{sign}(\mathbf{w}^T \mathbf{x}_i))\mathbf{x}_i$.

Outline

Some Theories

- Statistical Theory: studies how generalized a model is
- Learning Theory: explains how algorithms work
- Psychology: studies how computer can simulate human beings (neural networks)
- Optimization Theory: makes algorithms run faster

Cores of Statistical Theory

- Various statistical models (mostly on regression.)
- Each variable needs $k \geq 1$ samples to make it stable.

Cores of Learning Theory

- Extensive analysis on classification models.
- VC-dimension (for classification): complex models have poor generalizability
- Learning models for simple algorithms: online-learning, PAC-learning, SQ, etc.

Cores of Optimization Theory

- Hardness of the problem:
  $LP < QP < QCQP < SOCP < SDP$.
- CPU speed: x-GHz
- Memory access speed: DISK $\sim$ Network $<$ RAM $<$ Cache
- Example:

  3-GHz CPU, 100 clocks for 1kB memory access and in avg 100 clocks per meta-instruction, linear time algorithm, 2GB data points in memory: $\frac{2\times 100}{3} < 66(\times\text{convergence iterations})secs$

From Theories to Packages

- Consider computer architecture: embedded, multi-core or distributed?
- Consider what algorithms to support.
- Write docs (or build a discussion group) for target users!!!

From Theories to Packages

- Consider computer architecture: embedded, multi-core or distributed?
- Consider what algorithms to support.
- Write docs (or build a discussion group) for target users!!!

Let's see some example packages.

LIBSVM

- Optimized for multi-core architecture (using OPENMP).
- Linear to quadratic time algorithms to train non-linear models.
- Linear convergence: $O(1/\epsilon)$

LIBLINEAR

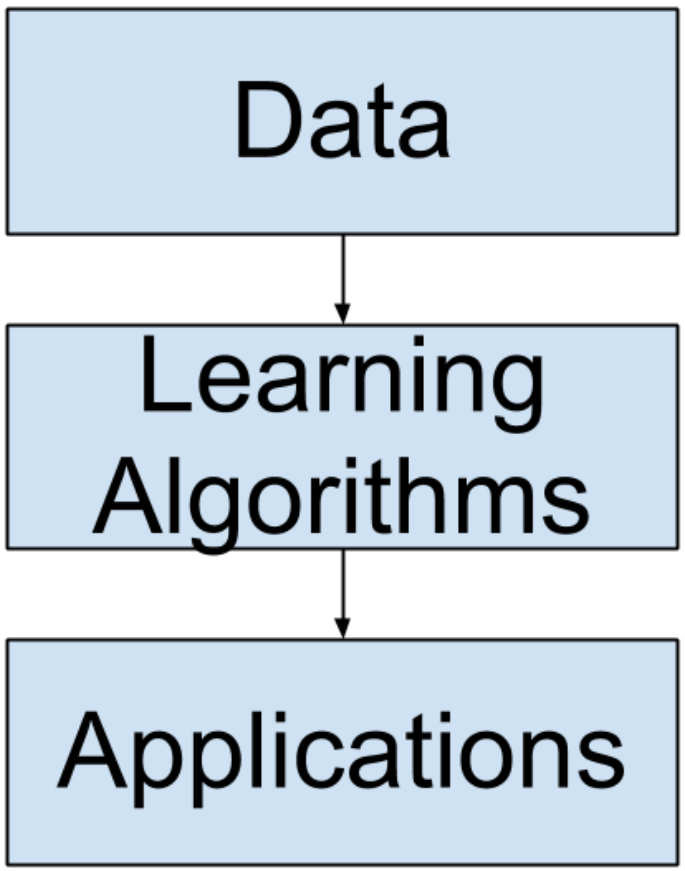
- Optimized for multi-core architecture (using OPENMP).
- Linear time algorithms to train linear models.
- Linear convergence: $O(1/\epsilon)$
- Application to text classification: LIBSHORTTEXT.

Take-home Message

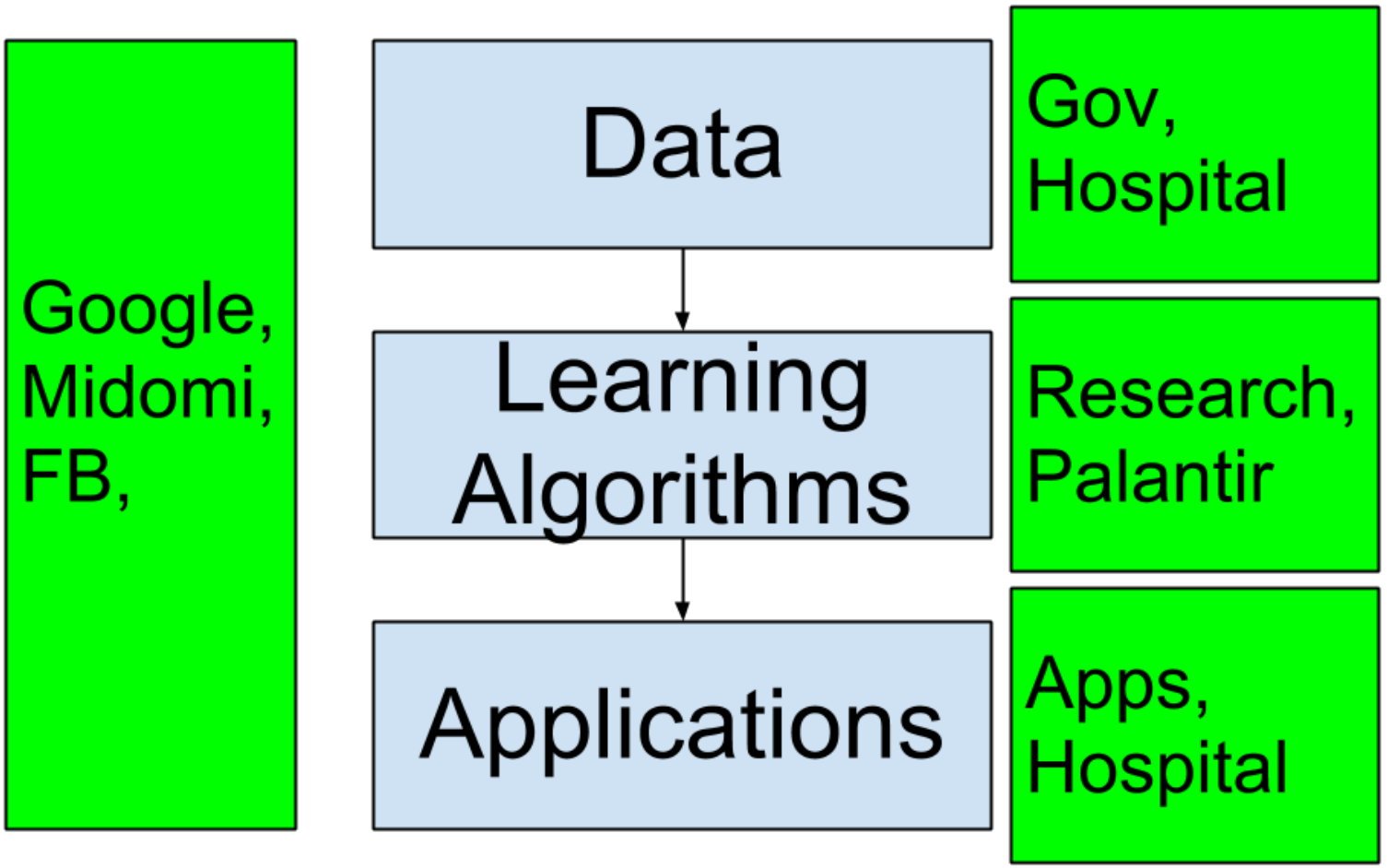
- Theories are mostly only for reference...
- Good implementation can overcome theoretical difficulties...
- But theoretical mindset is important.

Outline

A Real World Machine Learning System



A Real World Machine Learning System with Industries



Challenges of Data Storage

- Format: how to make other people access if needed
- Fast data access: No-SQL, SQL vs files on disk...
- Privacy concerns

Challenges of Model Complexity

- Simple model: fast training/prediction time, accountable, saving computational power
- Complex model: higher accuracy

Good Models in Practice

- Use meaningful features only
- Not sensitive to changes

Outline

Review

- Machine learning: learning from data
- Give example codes for generative models and discriminative models
- Introduces some theories and how they are used in practice
- Connect all these to the industry needs

Code to infinity and beyond! Thanks!

Thanks! The most update-to-date code and slides are at `https://github.com/scan33scan33/easyml`.