Large-Scale Machine Learning at Twitter
Large-Scale Machine Learning at Twitter

Jimmy Lin and Alek Kolcz

Twitter, Inc.
Large-Scale Machine Learning at Twitter

Outline

• Is twitter big data?
• How can machine learning help twitter?
• Existing challenges?

• Existing literature of large-scale learning
• Overview of machine learning
• Twitter analytic stack
• Extending pig

• Scalable machine learning
• Sentiment analysis application
Large-Scale Machine Learning at Twitter

What we will not talk about:

- Different “useful” application of twitter
- Why Twitter is a great product and one of its kind

What we will talk about:

- Challenges faced while making it a good product
- Solution approach by “Insiders”
The Scale of Twitter

- Twitter has more than 280 million active users
- 500 million Tweets are sent per day
- 50 million people log into Twitter every day
- Over 600 million monthly unique visitors to twitter.com

Large scale infrastructure of information delivery

- Users interact via web-ui, sms, and various apps
- Over 70% of our active users are mobile users
- Real-time redistribution of content
- At Twitter HQ we consume 1,440 hard boiled eggs weekly
- We also drink 585 gallons of coffee per week
Support for user interaction

• Search
  – Relevance ranking
• User recommendation
  – WTF or Who To Follow
• Content recommendation
  – Relevant news, media, trends

(other) problems we are trying to solve

• Trending topics
• Language detection
• Anti-spam
• Revenue optimization
• User interest modeling
• Growth optimization
Large-Scale Machine Learning at Twitter

To put learning formally...

**Supervised classification in a nutshell**

*Given*

\[ D = \left\{ (x_i, y_i) \right\}_i \]

*(sparse) feature vector*

*Induce*

\[ f : X \rightarrow Y \]

s.t. loss is minimized

\[
\text{empirical loss} = \frac{1}{n} \sum_{i=0}^{n} \ell(f(x_i), y_i)
\]

*loss function*

*Consider functions of a parametric form:*

\[
\arg \min_{\theta} \frac{1}{n} \sum_{i=0}^{n} \ell(f(x_i; \theta), y_i)
\]

*model parameters*

Key insight: machine learning as an optimization problem!
(closed form solutions generally not possible)
Literature

• Traditionally, the machine learning community has assumed sequential algorithms on data fit in memory (which is no longer realistic)
• Few publication on machine learning work-flow and tool integration with data management platform
  Google – adversarial advertisement detection
  Predictive analytic into traditional RDBMSes
  Facebook – business intelligence tasks
  LinkedIn – Hadoop based offline data processing
But they are not for machine learning specifically.
  Spark
  ScalOps
But they result in end-to-end pipeline.
Contribution

• Provided an overview of Twitter’s analytic stack
• Describe pig extension that allow seamless integration of machine learning capability into production platform
• Identify stochastic gradient descent and ensemble methods as being particularly amenable to large-scale machine learning

Note that,
No fundamental contributions to machine learning
Scalable Machine Learning

Scalable Machine learning

- Techniques for large-scale machine learning
- Stochastic gradient descent
- Ensemble method
Large-Scale Machine Learning at Twitter

Gradient Descent..
Large-Scale Machine Learning at Twitter

General method for nonlinear optimization

Start at \( \mathbf{w}(0) \); take a step along steepest slope

Fixed step size: \( \mathbf{w}(1) = \mathbf{w}(0) + \eta \hat{\mathbf{v}} \)

Next Weight = Current Weight + move
Move = Step Size \times Unit Vector

What is the direction \( \hat{\mathbf{v}} \)?

\( E_{\text{in}}(\mathbf{w}) \)

In-sample Error, \( E_{\text{in}} \)

Weights, \( \mathbf{w} \)

Slides from Yaser Abu Mostafa-Caltech
Large-Scale Machine Learning at Twitter

Gradient Descent...

Formula for the direction $\hat{v}$

\[
\Delta E_{in} = E_{in}(w(0) + \eta\hat{v}) - E_{in}(w(0))
\]

\[
= \eta \nabla E_{in}(w(0))^T\hat{v} + O(\eta^2)
\]

Using Taylor's series expansion

Because the surface non linear

\[
\geq -\eta \|\nabla E_{in}(w(0))\| \]

Since $\hat{v}$ is a unit vector,

\[
\hat{v} = -\frac{\nabla E_{in}(w(0))}{\|\nabla E_{in}(w(0))\|}
\]

Descent along gradient of error..

Slides from Yaser Abu Mostafa-Caltech
Large-Scale Machine Learning at Twitter

Stochastic Gradient Descent (SGD)

**Stochastic**

stəˈkastik/

*adjective*

1. randomly determined; having a random probability distribution or pattern that may be analyzed statistically but may not be predicted precisely.

**Stochastic gradient descent**

GD minimizes:

$$E_{in}(\mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} \frac{e^{(h(x_n), y_n)}}{\ln(1+e^{-y_n \mathbf{w}^T \mathbf{x}_n})}$$

← in logistic regression

by iterative steps along $-\nabla E_{in}$:

$$\Delta \mathbf{w} = -\eta \nabla E_{in}(\mathbf{w})$$

$\nabla E_{in}$ is based on all examples $(\mathbf{x}_n, y_n)$

“batch” GD

Slides from Yaser Abu Mostafa-Caltech
Stochastic gradient descent

Gradient Descent

\[ w^{(t+1)} = w^{(t)} + \gamma^{(t)} \frac{1}{n} \sum_{i=0}^{n} \nabla l (f(x_i; \theta^{(t)}), y_i) \]

Compute the gradient in the loss function by optimizing value in dataset. This method will do the iteration for all the data in order to one a gradient value.

Inefficient and everything in the dataset must be considered.
Stochastic gradient descent

Approximating gradient depends on the value of gradient for one instance.

\[ w^{(t+1)} = w^{(t)} + \gamma^{(t)} \nabla l \left( f(x; \theta^{(t)}), y \right) \]

Solve the iteration problem and it does not need to go over the whole dataset again and again.

Stream the dataset through a single reduce even with limited memory resource.

But when a huge dataset stream goes through a single node in cluster, it will cause network congestion problem.
Large-Scale Machine Learning at Twitter

Stochastic Gradient Descent (SGD)

Benefits of SGD

1. cheaper computation
2. randomization
3. simple

Rule of thumb: randomization helps
Large-Scale Machine Learning at Twitter

What is aggregation?

Combining different solutions $h_1, h_2, \cdots, h_T$ that were trained on $\mathcal{D}$:

Regression: take an average

Classification: take a vote

a.k.a. ensemble learning and boosting
Large-Scale Machine Learning at Twitter

Aggregation a.k.a Ensemble Learning

Different from 2-layer learning

In a 2-layer model, all units learn **jointly**:

![Diagram of a 2-layer model with units learning jointly]

In aggregation, they learn **independently** then get combined:

![Diagram of an aggregation model with units learning independently]

Slides from Yaser Abu Mostafa-Caltech
Ensemble Methods

Classifier ensembles: high performance learner

Performance: very well

Some rely mostly on randomization
- Each learner is trained over a subset of features and/or instances of the data

Ensembles of linear classifiers

Ensembles of decision trees (random forest)
Large-Scale Machine Learning at Twitter

AVOID: “jankky” analysis of messy data

- Too Much Data
- Down-Sample
- Run Model on all data
- Build model on a local box
- Probably lose intermediate results and data
In a big sample (large \(N\)), \(\nu\) is probably close to \(\mu\) (within \(\epsilon\)).

Formally,

\[
P\left[|\nu - \mu| > \epsilon\right] \leq 2e^{-2\epsilon^2 N}
\]

Sample frequency \(\nu\) is likely lose to bin frequency \(\mu\).

This is called Hoeffding’s Inequality.
Large-Scale Machine Learning at Twitter

Apache Hadoop Ecosystem

Hadoop Ecosystem

Big Table open source version

Image Source: Apache Yarn Release
Large-Scale Machine Learning at Twitter

Hadoop Ecosystem at Twitter...

Oink:
- Aggregation query
  Standard business intelligence tasks
- Ad hoc query
  One-off business request
- Prototypes of new function
  Experiment by analytic group

Hadoop cluster

Database

Application log

Other sources

Real-time processes

Batch processes

Serialization
Protocol buffer/Thrift

HDFS
Why use Pig?

- Suppose you have user data in one file, website data in another, and you need to find the top 5 most visited sites by users aged 18 - 25
Large-Scale Machine Learning at Twitter

In Map-Reduce

170 lines of code, 4 hours to write

Credits: Hortonworks
Large-Scale Machine Learning at Twitter

In Pig Latin

Users = load 'input/users' using PigStorage(',,') as (name:chararray, age:int);
Fltrd = filter Users by age >= 18 and age <= 25;
Pages = load 'input/pages' using PigStorage(',,') as (user:chararray, url:chararray);
Jnd = join Fltrd by name, Pages by user;
Grpd = group Jnd by url;
Smmd = foreach Grpd generate group,COUNT(Jnd) as clicks;
Srtgd = order Smmd by clicks desc;
Top5 = limit Srtgd 5;
store Top5 into 'output/top5sites' using PigStorage(',,');

9 lines of code, 15 minutes to write

170 lines to 9 lines of code

Credits: Hortonworks
Maximizing the use of Hadoop

• We cannot afford too many diverse computing environments
• Most of analytics job are run using Hadoop cluster
  – Hence, that’s where the data live
  – It is natural to structure ML computation so that it takes advantage of the cluster and is performed close to the data

Seamless scaling to large datasets

Integration into production workflows
Large-Scale Machine Learning at Twitter

Core libraries:

Core Java library
Basic abstractions similar to existing packages (weka, mallet, mahout)

Lightweight wrapper
Expose functionalities in Pig
Training models:

```java
training = load 'training.txt'
    using SVMLightStorage()
    as (target: int, features: map[]);

store training into 'model/
    using FeaturesLRCClassifierBuilder();
```

Storage function
Large-Scale Machine Learning at Twitter

Shuffling data:

```
training = foreach training generate label, features, RANDOM() as random;
training = order training by random parallel 1;

data = foreach data generate target, features,
      RANDOM() as random;
split data into training if random <= 0.9,
  test if random > 0.9;
```
Using models:

```java
define Classify ClassifyWithLR('model/');
data = load 'test.txt' using SVMLightStorage()
as (target: double, features: map[]);
data = foreach data generate target,
Classify(features) as prediction;
```
Large-Scale Machine Learning at Twitter

Credits: Hortonworks
Final Learning - Ensemble Methods
Example: Sentiment Analysis

Emotion Trick 😊 😞

Test dataset: 1 million English tweets, minimum 20 letters-long

Training data: 1 million, 10 million and 100 million English training examples

Preparation: training and test sets contains equal number of positive and negative examples, removed all emoticons.
Large-Scale Machine Learning at Twitter

Finally a graph ..
1. The error bar denotes 95% confidence interval
2. The leftmost group of bars show accuracy when training a single logistic regression classifier on \{1, 10, 100\} million training examples.
3. 1-10 Change Sharp, 10 – 100 million: Not that sharp
4. The middle and right group of bars in Figure 2 show the results of learning ensembles
5. Ensembles lead to higher accuracy—and note that an ensemble trained with 10 million examples outperforms a single classifier trained on 100 million examples
6. No accurate running time reported as experiments were run on production clusters – but informal observations are in sync with what the logical mind suggests (ensemble takes shorter to train because models are learned in parallel)
7. In terms of applying the learned models, running time increases with the size of the ensembles—since an ensemble of \(n\) classifiers requires making \(n\) separate predictions.
What I loved about paper : I understood it 😊?

“our goal has never been to make fundamental contributions to machine learning, we have taken the pragmatic approach of using off-the-shelf toolkits where possible. Thus, the challenge becomes how to incorporate third-party software packages along with in-house tools into an existing workflow”.
Large-Scale Machine Learning at Twitter

ONE DOES NOT SIMPLY
END A PRESENTATION WITHOUT ANSWERING QUESTIONS
Large-Scale Machine Learning at Twitter

This is the end of the presentation

Clap now