DeepDive:
A Data System for Macroscopic Science

Jaeho Shin
(on behalf of Christopher Ré)
Stanford University
Computer Science
http://deepdive.stanford.edu/

Dark Data System.

Think about **features** not **algorithms**.

Why, What, and How
First, a story...
Newton

Principia Mathematica
Turns out, Newton was not alone
Millennia of Knowledge in Libraries
Now: freely available & digital
The world’s scientific knowledge is accessible.
But we’re still human...
The world’s scientific knowledge is accessible, but not readable.
Today, some pressing problems require macroscopic knowledge.

- Climate & Biodiversity
- Health
- Financial Markets
Could we build a machine to \textbf{read} for us?
Data are buried in tables, but not in a self-contained way.
Data are buried in tables, but not in a self-contained way.
PaleoDeepDive

Shanan Peters (Geo) and Miron Livny (CS)  
DeepDive.Stanford.edu (Ce Zhang et al.)
T. Rex are found dating to the upper Cretaceous.

Statistical Inference

Appears("T. Rex", "Cretaceous")

Aggressive Approach

Every character, word, part of speech is a variable statistical inference on billions of variables.
PaleoDB

- Human-created
- 329 volunteers
- 13 years
- 46K documents
- 200+ Papers, 17 Nature/Science

PaleoDeepDive

- Machine-created
- 10x documents
- 100x extractions

Preliminary Precision

- PaleoDB Volunteers: 0.84
- PaleoDeepDive: 0.94

Hope: knowledge bases can help **accelerate science**.

*Tree of Life*  
*Drug Repurposing*  
*Genomics*

*Used by a number of companies.*  
*A Dark Data System*
Human Trafficking on the (Dark) Web...

Hypothesis: Trafficked individuals offer lower cost and riskier sexual services.

Plain sight: Open/Dark Web ads for such services

Challenges:
1. Need high-resolution information to build model. (rates, ethnicity, location, etc.)
2. Scientific papers are clear, dark web obfuscated.
Human Trafficking on the (Dark) Web...

Open & Dark Web (ad-style text). JPL et al.

Structured Info: Phone #, Rates of Service, ...

Use: law enforcement/NGOs
Quality of data

For DARPA MEMEX, we recently (< 6 months)

- Processed >35M advertisement-style documents
- Yielded ~26M structured records
- Tens of columns (location, phone #, price, etc)
- With compute times of less than a day.

Incremental (VLDB15)

<table>
<thead>
<tr>
<th>Field</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>price</td>
<td>95%</td>
<td>90%</td>
</tr>
<tr>
<td>isSexAd</td>
<td>100%</td>
<td>97%</td>
</tr>
<tr>
<td>location</td>
<td>91%</td>
<td>88%</td>
</tr>
<tr>
<td>name</td>
<td>84%</td>
<td>87%</td>
</tr>
<tr>
<td>phone</td>
<td>100%</td>
<td>98%</td>
</tr>
<tr>
<td>requirement</td>
<td>83%</td>
<td>50%</td>
</tr>
</tbody>
</table>

New York DA uses MEMEX for all HT investigations this year.
Think about features not algorithms.

http://deepdive.stanford.edu/
DeepDive: It’s all features!

Task
Every character, word, part of speech is a variable
Statistical Inference on billions of variables.

User tells us ...
- (incomplete) dictionaries of places,
- How to resolve places to lat-longs,
- words that indicate a relationship

How does a user do it?
Example: Extracting Spouse Relations

Corpus (Dark Data)
U.S. President Barack Obama's wife Michelle Obama honored all mothers on Mother's Day and offered her thoughts ...

How do we produce tuples like

\text{Married}(Barack Obama, Michelle Obama)

And all other married couples in text?

Examples on Github!
U.S. President Barack Obama’s wife Michelle Obama honored all mothers on Mother’s Day and offered her thoughts ...

**Corpus (Dark Data)**

**text**

U.S. President Barack Obama’s wife Michelle Obama honored all mothers on Mother’s Day and offered her thoughts ...

**Sentences**

<table>
<thead>
<tr>
<th>words</th>
<th>POS</th>
<th>NER</th>
<th>SID</th>
</tr>
</thead>
<tbody>
<tr>
<td>[U.S.,President,Barack,Obama,’s,wife,Michelle,Obama,...]</td>
<td>[NNP,NNP,NNP,NNP,...]</td>
<td>[LOC,O,PER,PER,...]</td>
<td>S1</td>
</tr>
</tbody>
</table>

**DDlog: Declarative** inspired by Datalog, MLNs, Tuffy.

*In 0.6.0 release*
1. Candidate Mappings

## Sentences

<table>
<thead>
<tr>
<th>words</th>
<th>POS</th>
<th>NER</th>
<th>SID</th>
</tr>
</thead>
<tbody>
<tr>
<td>[U.S., President, Barack, Obama, 's, wife, Michelle, Obama, ...]</td>
<td>[NNP, NNP, NNP, NNP, POS, NN, NNP, NNP, NNP, ...]</td>
<td>[LOC, O, PER, PER, R, O, O, PER, PER, ...]</td>
<td>S1</td>
</tr>
</tbody>
</table>

### Mentions

```logic
Mentions :- !ext_person(Sentences).
```

```python
function ext_person over (words, pos, ner, sid)
    returns (sid, mid, words)
implementation "udf/find_person.py".
```

### MarriedCandidate

```logic
MarriedCandidate(s, p1, p2) :-
    Mentions(s, p1, _), Mentions(s, p2, _).
```

### Mentions

<table>
<thead>
<tr>
<th>SID</th>
<th>MID</th>
<th>words</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>M1</td>
<td>[Barack, Obama]</td>
</tr>
<tr>
<td>S1</td>
<td>M2</td>
<td>[Michelle, Obama]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MarriedCandidate</th>
<th>SID</th>
<th>MID</th>
<th>MID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S1</td>
<td>M1</td>
<td>M2</td>
</tr>
</tbody>
</table>
2. Feature Extraction

**Sentences**

<table>
<thead>
<tr>
<th>words</th>
<th>POS</th>
<th>NER</th>
<th>SID</th>
</tr>
</thead>
<tbody>
<tr>
<td>[U.S.,President,Barack,Obama,’s,wife,Michelle,Obama,...]</td>
<td>[NNP,NNP,NNP,NNP,NNP,POS,NN,NNP,NNP,...]</td>
<td>[LOC,O,PER,PER,O,O,PER,PER,PER,...]</td>
<td>S1</td>
</tr>
</tbody>
</table>

**Features** :- !ext_features(Sentences, MarriedCandidate).

function ext_features ...

implementation "udf/ext_features.py".

**Mentions**

<table>
<thead>
<tr>
<th>SID</th>
<th>MID</th>
<th>words</th>
<th>MarriedCandidate</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>M1</td>
<td>[Barack,Obama]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1</td>
<td>M2</td>
<td>[Michelle,Obama]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>MID</td>
<td>MID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>S1</td>
<td>M1</td>
</tr>
</tbody>
</table>
3. Inference Rules

Married(p1,p2) :-
  MarriedCandidate(_,p1,p2),
  Features(p1,p2,f)
weight = f.

Just defined a Binary classifier!

Married is an (incomplete) set of examples
Users write Features & Transformation in DDlog (Inspired by MLNs) & Python.

Sentences :- !nlp(Corpus).
function nlp over (text) returns (words,pos,ner,sid)
  implementation "udf/corenlp.sh ..." .

Mentions :- !ext_person(Sentences).
function ext_person over (words,pos,ner,sid) returns (sid,mid,words)
  implementation "udf/find_person.py".
MarriedCandidate(s,p1,p2) :- Mentions(s,p1,_), Mentions(s,p2, _).

Features :- !ext_features(Sentences, MarriedCandidate).
function ext_features ... implementation "udf/ext_features.py".

Married(p1,p2) :- MarriedCandidate(_,p1,p2), Features(p1,p2,f)
weight = f.

No reference to algorithms: Just features. DD does the rest. (Demos online)
How do you debug?
Calibration plots.
Meaningful probability, not just scores.

- **Ideal** line represents the perfect accuracy where the output probability matches the actual probability.
- **Actual** line shows the real-world accuracy, often deviating from the ideal due to various factors.
- The **BelongsTo(Taxon, Taxon)** box highlights examples where the model correctly identifies species:
  - (Autoceta, Cetacea)
  - (Aulophysester, Hoplocetinae)
  - (Acantharodeia, Dasypodidae)
  - (Acdestis, Caenolestidae)
- A notable exception is (Castoridae, Boreofiber), indicated by an **X**.
Candidates for improvement

Output to users

# Extractions

Goal
Suggests it may be possible to think about features not algorithms using marginals
Broadly Usable

- **PaleoDeepDive**
  - 2 years
  - 1 CS Student

- **PharmaDeepDive**
  - 6 months
  - 1 BioE Student

How do we make building a KB easier and cheaper?

Think about **features** not **algorithms**.

A **framework** for feature engineering.

[SIGMOD14, Best Paper: ADMM + DSL to train 100 models as quickly as 1 & DEBUL14]
To get rid of algorithms, we need a fast engine....

Let hardware do the work.
Staggering amount of machine learning/stats can be written as:

$$\min_x \sum_{i=1}^{N} f(x, y_i)$$

N (number of $y_i$s, data) typically in the billions

Ex: Classification, Recommendation, Deep Learning.

**De facto** iteration to solve large-scale problems: **SGD**.

$$x^{k+1} = x^k - \alpha N \nabla f(x^k, y_j)$$

Select one term, j, and estimate gradient.

Billions of tiny iterations.
How do we run SGD in Parallel?

Data Systems Perspective of SGD.

\[ x^{k+1} = x^k - \alpha N \nabla f(x^k, y_j) \]

**Insane conflicts:** Billions of tiny (~100 instructions) jobs, RW conflicts on \(x\), which is called the model.

How can we hope to speed this up with parallelism?

Serializability seems hopeless…
How do we run SGD in Parallel?

**Thm:** If we do *no locking*, SGD still converges to right answer—at essentially the same theoretical rate!

**Hogwild!** [Niu, Recht, Ré, Wright NIPS11]
**AsySCD (Dense)** [Liu, Wright et al. ICML14, JMLR14]
**Nonconvex Power Methods** [DeSa & Ré 15]

Technical conditions on ratio of processors, delays, (semantic) sparsity.

**High-level idea:** answer is only statistically correct.
A larger trend?

*NB:* There is theory here SGD [NIPS11], SCD [ICML14], more soon and systems work [SIGMOD13, SIGMOD14, VLDB14]

Relaxing **consistency** to be **architecturally aware** can be a big performance win.
Cortana: Microsoft’s Digital Assistant

AI breakthrough: Microsoft’s ‘Project Adam’ identifies dog breeds, points to future of machine learning

All web companies have similar: image rec, voice, mobile, search, etc.

“...using a technology called, of all things, Hogwild!”

http://www.wired.com/2014/07/microsoft-adam/
The Key Balance

Key Issue: Balance Statistical versus Hardware Efficiency.

Statistical versus Hardware Efficiency

Relaxing consistency results in new tradeoffs.

1. Access methods
   - {Row, Column, Row-col}
2. Model Replication
   - {Core, Node, Machine}
3. Data Replication
   - {Full, Importance, Shard}

Can be 100x faster than classical choices, e.g., MLlib or GraphLab

Future Work: *Image and Textual Knowledge bases*
On-going Work: *Image and Textual Knowledge bases*

Images are key source of *dark data* in applications.
Go Beyond Text Processing...

Images are important to many scientific questions

<table>
<thead>
<tr>
<th>What kind of dinosaur is this?</th>
<th>Does this patient have short fingers?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is this sea star sick?</td>
<td>Is this lung cancer?</td>
</tr>
</tbody>
</table>

Can we extend DeepDive to support image processing algorithms?
DeepDive meets Image Processing

Recognizing Different Critters

Can we teach our machine to recognize taxonomy?
One Proof-of-concept Example

“read” a Paleontology textbook and learn the difference between sponges and shells?

Use Distant Supervision!
Comparable to a Human?

DeepDive Extractions: High quality already!

**Figure Name Mention** → **Taxon Mention**

Fig. 387, 1a-c. *B. rara*. Serpukhovian, Kazakhstan, Dzhezgazgan district; a, b, holotype, viewed ventrally, laterally, MGU 31/342, XI (Litvinovich, 1967);

Figures

Test with Human Labels

- 3K Brachiopoda Images
- 2K Porifera Images

**Accuracy = 94%**
Vision + DeepDive?

Deep Learning with CNNs

*High quality with little or no user involvement*

CNNs are awesome! But...

- 😞 CNNs use thousands of machine hours to train.
- 😞 The extracted features are difficult for scientists to interpret.
- 😞 CNNs are hard for the user to debug in complex applications.

*Do we have a fast, easy-to-explain, alternative?*
How does one represent images in a microscope or telescope?

F. Zernike

Zernike Polynomials

1. Interpret with decades of optics.
2. Superfast algorithms \((N \log^2 N)\)
Fossil Classification with Zernike Transforms

Porifera

Brachiopoda

Use (multiresolution) Zernike transform to get the coefficients

Train SVM

CNN Accuracy

Accuracy = 94%

Zernike Accuracy

Accuracy = 95%

Can we apply the same trick more broadly?

MNIST: Better. ImageNet: Not Awful…
Slightly deeper fusion of images and text.
Application: Early-stage Lung Cancer

Currently done with stage information

Images & pathology notes to predict outcomes of lung cancer patients?

Credit: Kun-Hsing Yu, Daniel Rubin, Michael Snyder (manuscript in preparation)
Some Early Results

Pathology Images

Visual Features

Probability of Survival

Cleanly separated

Pathology Reports

Text Extractions

A. Histologic type:
Adenocarcinoma
Histologic grade:
Moderately differentiated

A. Histologic type:
Adenocarcinoma
Histologic grade:
Moderately

Deep Dive
Conclusion

1. KBs help with macroscopic questions

2. Probabilistic inference = algorithmic independence

3. Hardware-aware engine is our current approach