Injecting Logical Background Knowledge into Embeddings for Relation Extraction
Tim Rocktäschel, Sameer Singh and Sebastian Riedel
t.rocktaschel@cs.ucl.ac.uk

2nd of June 2015
NAACL, Denver
Freebase is incomplete

• **Missing Facts:** `placeOfBirth` attribute is missing for 71% of the people (Dong et al., 2014)

• **Missing Entities:** Contains no information about UCL Machine Reading Lab

• **Missing Relations:** May contain `profAt(John Shawe-Taylor, UCL)` but not `givesLecturesAt(John Shawe-Taylor, UCL)`

• **Machine reading and reasoning to the rescue!**
Relation Extraction with Matrix Factorization and Universal Schemas (Riedel et al., 2013)
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

<table>
<thead>
<tr>
<th>Textual Patterns</th>
<th>Freebase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Petrie, UCL</td>
<td></td>
</tr>
<tr>
<td>Ferguson, Harvard</td>
<td></td>
</tr>
<tr>
<td>Andrew, Cambridge</td>
<td></td>
</tr>
<tr>
<td>Trevelyan, Cambridge</td>
<td></td>
</tr>
</tbody>
</table>
Relation Extraction with Matrix Factorization and Universal Schemas (Riedel et al., 2013)
### Relation Extraction with Matrix Factorization and Universal Schemas

(Riedel et al., 2013)

**Textual Patterns vs. Freebase**

<table>
<thead>
<tr>
<th>Petrie, UCL</th>
<th>1.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ferguson, Harvard</td>
<td>?</td>
</tr>
<tr>
<td>Andrew, Cambridge</td>
<td>?</td>
</tr>
<tr>
<td>Trevelyan, Cambridge</td>
<td>?</td>
</tr>
</tbody>
</table>
Relation Extraction with Matrix Factorization and Universal Schemas (Riedel et al., 2013)
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

<table>
<thead>
<tr>
<th></th>
<th>Petrie, UCL</th>
<th>Ferguson, Harvard</th>
<th>Andrew, Cambridge</th>
<th>Trevelyan, Cambridge</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-is-historian-at-Y</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>?</td>
</tr>
<tr>
<td>X-is-professor-at-Y</td>
<td>1.00</td>
<td></td>
<td>1.00</td>
<td>?</td>
</tr>
<tr>
<td>X-museum-at-Y</td>
<td></td>
<td></td>
<td></td>
<td>?</td>
</tr>
<tr>
<td>X-teaches-history-at-Y</td>
<td></td>
<td></td>
<td></td>
<td>?</td>
</tr>
<tr>
<td>employeeAt(X, Y)</td>
<td></td>
<td></td>
<td></td>
<td>?</td>
</tr>
</tbody>
</table>

Textual Patterns | Freebase
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

<table>
<thead>
<tr>
<th></th>
<th>Petrie, UCL</th>
<th>Ferguson, Harvard</th>
<th>Andrew, Cambridge</th>
<th>Trevelyan, Cambridge</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Textual Patterns</strong></td>
<td>?</td>
<td>1.00</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>X-is-historian-at-Y</td>
<td>1.00</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>X-is-professor-at-Y</td>
<td>1.00</td>
<td>1.00</td>
<td>?</td>
<td>1.00</td>
</tr>
<tr>
<td>X-museum-at-Y</td>
<td>?</td>
<td>?</td>
<td>1.00</td>
<td>?</td>
</tr>
<tr>
<td>employeeAt(X,Y)</td>
<td>?</td>
<td>1.00</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>

**Freebase**
Relation Extraction with Matrix Factorization and Universal Schemas (Riedel et al., 2013)
Relation Extraction with Matrix Factorization and Universal Schemas (Riedel et al., 2013)

\[
\begin{array}{ccc}
\text{Petrie, UCL} & 1.00 & 1.00 & 1.00 \\
\text{Ferguson, Harvard} & 1.00 & 1.00 & \text{?} \\
\text{Andrew, Cambridge} & 1.00 & 1.00 & \text{?} \\
\text{Trevelyan, Cambridge} & 1.00 & 1.00 & \text{?} \\
\end{array}
\]

\[\mathbf{v}_r \in \mathbb{R}^k\]
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

<table>
<thead>
<tr>
<th></th>
<th>Petrie, UCL</th>
<th>Ferguson, Harvard</th>
<th>Andrew, Cambridge</th>
<th>Trevelyan, Cambridge</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-is-historian-at-Y</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>X-is-professor-at-Y</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X-museum-at-Y</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X-teaches-history-at-Y</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>employeeAt(X, Y)</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ \mathbf{v}_{r1}, \mathbf{v}_{r2}, \mathbf{v}_{r3}, \mathbf{v}_{r4}, \mathbf{v}_{r5} \in \mathbb{R}^{k} \]
Relation Extraction with Matrix Factorization and Universal Schemas (Riedel et al., 2013)

\[ X:\text{is-historian-at-} Y \]
\[ X:\text{is-professor-at-} Y \]
\[ X:\text{museum-at-} Y \]
\[ X:\text{teaches-history-at-} Y \]
\[ \text{employeeAt}(X, Y) \]

\begin{tabular}{|c|c|c|c|c|}
\hline
 & Petrie, UCL & Ferguson, Harvard & Andrew, Cambridge & Trevelyan, Cambridge \\
\hline
\hline
0.06 & 0.97 & 0.93 & 0.07 & 0.96 \\
\hline
0.93 & 0.94 & 0.03 & 0.06 & 0.88 \\
\hline
0.00 & 0.95 & 0.10 & 0.95 & 0.76 \\
\hline
0.96 & 0.03 & 0.00 & 0.06 & 0.96 \\
\hline
\end{tabular}

\[ \mathbf{v}_{r_1}, \mathbf{v}_{r_2}, \mathbf{v}_{r_3}, \mathbf{v}_{r_4}, \mathbf{v}_{r_5} \in \mathbb{R}^k \]
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

<table>
<thead>
<tr>
<th></th>
<th>Petrie, UCL</th>
<th>Ferguson, Harvard</th>
<th>Andrew, Cambridge</th>
<th>Trevelyan, Cambridge</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-is-historian-at-Y</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X-is-professor-at-Y</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X-museum-at-Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X-teaches-history-at-Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>employeeAt(X, Y)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ \mathbf{v}_i \in \mathbb{R}^k \]
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

**Open Information Extraction**

<table>
<thead>
<tr>
<th>Role</th>
<th>Petrie, UCL</th>
<th>Ferguson, Harvard</th>
<th>Andrew, Cambridge</th>
<th>Trevelyan, Cambridge</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-is-historian-at-Y</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>X-is-professor-at-Y</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X-museum-at-Y</td>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>X-teaches-history-at-Y</td>
<td></td>
<td></td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>EmployeeAt(X, Y)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ \mathbf{v}_{r_1}, \mathbf{v}_{r_2}, \mathbf{v}_{r_3}, \mathbf{v}_{r_4}, \mathbf{v}_{r_5} \in \mathbb{R}^k \]
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

**Distant Supervision**

<table>
<thead>
<tr>
<th></th>
<th>Petrie, UCL</th>
<th>Ferguson, Harvard</th>
<th>Andrew, Cambridge</th>
<th>Trevelyan, Cambridge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
\mathbf{v}_{p_1} & \quad \mathbf{v}_{p_2} \\
\mathbf{v}_{p_3} & \quad \mathbf{v}_{p_4} \\
\end{align*}
\]

\[
\mathbf{v}_{r_1} \quad \mathbf{v}_{r_2} \quad \mathbf{v}_{r_3} \quad \mathbf{v}_{r_4} \quad \mathbf{v}_{r_5} \\
\in \mathbb{R}^k
\]
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

**Distant Supervision**

Fails for no or little alignment!

<table>
<thead>
<tr>
<th></th>
<th>Petrie, UCL</th>
<th>Ferguson, Harvard</th>
<th>Andrew, Cambridge</th>
<th>Trevelyan, Cambridge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

\[
\mathbf{v}_{\rho_1} \quad \mathbf{v}_{\rho_2} \quad \mathbf{v}_{\rho_3} \quad \mathbf{v}_{\rho_4}
\]

\[
\mathbf{v}_{r_1} \quad \mathbf{v}_{r_2} \quad \mathbf{v}_{r_3} \quad \mathbf{v}_{r_4} \quad \mathbf{v}_{r_5}
\]

\[
\in \mathbb{R}^k
\]
Relation Extraction with Matrix Factorization and Universal Schemas *(Riedel et al., 2013)*

**Representation Learning**

<table>
<thead>
<tr>
<th>Relation</th>
<th>Matrix Representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Petrie, UCL</td>
<td><img src="matrix.png" alt="Matrix for Petrie, UCL" /></td>
</tr>
<tr>
<td>Ferguson, Harvard</td>
<td><img src="matrix.png" alt="Matrix for Ferguson, Harvard" /></td>
</tr>
<tr>
<td>Andrew, Cambridge</td>
<td><img src="matrix.png" alt="Matrix for Andrew, Cambridge" /></td>
</tr>
<tr>
<td>Trevelyan, Cambridge</td>
<td><img src="matrix.png" alt="Matrix for Trevelyan, Cambridge" /></td>
</tr>
</tbody>
</table>

\[
\begin{pmatrix}
    v_{r1} & v_{r2} & v_{r3} & v_{r4} & v_{r5} \\
    v_{p1} & v_{p2} & v_{p3} & v_{p4}
\end{pmatrix} \in \mathbb{R}^k
\]
Relation Extraction with Matrix Factorization and Universal Schemas (Riedel et al., 2013)

Representation Learning
Hard to fix mistakes!
We can fix this with logical formulae!

- **Representation Learning**: hard to fix mistakes
  \[ \neg \text{person}(x) \Rightarrow \neg \text{place of birth}(x, y) \]

- **Distant Supervision**: fails for no or little alignment
  \[ \text{#1-is-a-student-in-#2-'s-lab}(x, y) \Rightarrow \text{supervisedBy}(x, y) \]

- **Pros**: Formulae are easy to modify and improve
- **Cons**: Brittle, no generalization and inference can become intractable
- **Markov Logic Networks**: easy to modify, generalize well, but inference often intractable in practice
Overview
Overview
Overview

Evidence   Sparse Training Matrix

|\mathcal{P}|   |\mathcal{R}|
Overview

Evidence $\rightarrow$ Sparse Training Matrix $\rightarrow$ Matrix Factorization

$|P|$ $\rightarrow$ Facts $\rightarrow$ $|P|$ $k$

$|R|$ $k$
Overview

Evidence → Sparse Training Matrix

| \( \mathcal{P} \) |

| \( \mathcal{R} \) |
Overview

Evidence ➔ Sparse Training Matrix

∀x, y : co-founder-of(x, y) ⇒ company/founders(y, x)
∀x, y : review-by(x, y) ⇒ author/works_written(y, x)
∀x, y : daughter-of(x, y) ⇒ person/parents(x, y)
Overview

Evidence → Sparse Training Matrix → Low-rank Logic Embeddings

∀x, y: co-founder-of(x, y) ⇒ company/founders(y, x)
∀x, y: review-by(x, y) ⇒ author/works_written(y, x)
∀x, y: daughter-of(x, y) ⇒ person/parents(x, y)

Facts

First-order Formulae
Overview

Evidence → Sparse Training Matrix

\( |\mathcal{R}| \)

Facts

First-order Formulae

\( \forall x, y : \text{co-founder-of}(x, y) \Rightarrow \text{company/founders}(y, x) \)
\( \forall x, y : \text{review-by}(x, y) \Rightarrow \text{author/works_written}(y, x) \)
\( \forall x, y : \text{daughter-of}(x, y) \Rightarrow \text{person/parents}(x, y) \)

Low-rank Logic Embeddings → Completed Matrix

\( |\mathcal{P}| \)
Ways to combine matrix factorization and logic
Ways to combine matrix factorization and logic

\[ X \text{-is-professor-at-} Y \Rightarrow \text{employeeAt}(X, Y) \]
Ways to combine matrix factorization and logic

$X$-is-professor-at-$Y$ \implies employeeAt$(X, Y)$

<table>
<thead>
<tr>
<th>Riedel, UCL</th>
<th>Vlachos, Sheffield</th>
<th>Pontil, UCL</th>
<th>Petrie, UCL</th>
<th>Ferguson, Harvard</th>
<th>Andrew, Cambridge</th>
<th>Trevelyan, Cambridge</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Gray cells representing training facts" /></td>
<td><img src="image" alt="White cells representing predicted facts" /></td>
<td><img src="image" alt="Gray cells representing training facts" /></td>
<td><img src="image" alt="Gray cells representing training facts" /></td>
<td><img src="image" alt="Gray cells representing training facts" /></td>
<td><img src="image" alt="Gray cells representing training facts" /></td>
<td><img src="image" alt="Gray cells representing training facts" /></td>
</tr>
</tbody>
</table>

Legend:
- **Gray cells**: Training facts
- **White cells**: Predicted facts
Ways to combine matrix factorization and logic

\[
\text{X-is-professor-at-} \ Y \implies \\
\text{employeeAt(} X, \ Y \text{)}
\]
Ways to combine matrix factorization and logic

\( \mathbf{X} \)-is-professor-at-\( \mathbf{Y} \) ⇒
employeeAt(\( \mathbf{X}, \mathbf{Y} \))
Ways to combine matrix factorization and logic

\[
X\text{-is-professor-at}-Y \Rightarrow \text{employeeAt}(X, Y)
\]
Ways to combine matrix factorization and logic

\[ X \text{-is-professor-at-} Y \Rightarrow \text{employeeAt}(X, Y) \]
Ways to combine matrix factorization and logic

\[ \text{X-is-professor-at-Y} \Rightarrow \text{employeeAt(X, Y)} \]
Ways to combine matrix factorization and logic

\[ X \text{-is-professor-at-} Y \Rightarrow \text{employeeAt}(X, Y) \]
Ways to combine matrix factorization and logic

\[ X \text{-is-professor-at-} Y \Rightarrow \text{employeeAt}(X, Y) \]
Ways to combine matrix factorization and logic

$X$-is-professor-at-$Y$ $\Rightarrow$
employeeAt($X$, $Y$)

Diagram showing different entities and relations with color-coding for training facts, predicted facts, matrix factorization, logical inference, and post-factorization inference.
Ways to combine matrix factorization and logic

$X$-is-professor-at-$Y \Rightarrow$ employeeAt($X$, $Y$)


Legend:
- Training facts
- Predicted facts
- Matrix Factorization
- Logical Inference
- Post-Factorization Inference
Ways to combine matrix factorization and logic

$X$-is-professor-at-$Y \Rightarrow$
employeeAt$(X, Y)$
Ways to combine matrix factorization and logic

\[ \text{X-is-professor-at-} \Rightarrow \text{Y} \Rightarrow \text{employeeAt}(X, Y) \]
Ways to combine matrix factorization and logic

X-is-professor-at-Y ⇒ employeeAt(X, Y)

- X’s-lab-at-Y
- X-gives-lectures-at-Y
- X-museum-at-Y
- affiliatedWith(X, Y)

Legend:
- Training facts
- Predicted facts
- Matrix Factorization
- Logical Inference
- Post-Factorization Inference
- Pre-Factorization Inference

Riedel, UCL
Vlachos, Sheffield
Pontil, UCL
Petrie, UCL
Ferguson, Harvard
Andrew, Cambridge
Trevelyan, Cambridge

Premise    Consequent
Ways to combine matrix factorization and logic

\[ X \text{-is-professor-at-} Y \Rightarrow \]
\[ \text{employeeAt}(X, Y) \]
Injecting Logical Background Knowledge
Matrix factorization training objective

- Probability of \( r_s(e_i, e_j) \) being true
  \[
  p(r_s(e_i, e_j) = \text{true}) := \sigma(v_s \cdot v_{ij})
  \]

- Log-likelihood loss
  \[
  \max_{\{v_s\},\{v_{ij}\}} \sum_{(s,i,j) \in T} \log \left( \sigma(v_s \cdot v_{ij}) \right) + \sum_{(s,u,w) \notin T} \tau_{suw} \log (1 - \sigma(v_s \cdot v_{uw}))
  \]

- Optimized using AdaGrad (Duchi et al., 2011)
Matrix factorization is "injecting" atomic formulae

- Training facts are ground atoms: \( \mathcal{F} = r_s(e_i, e_j) \)
- Let \([\cdot]\) denote mapping from symbolic formulae to expectations
- \([r_s(e_i, e_j)] := \sigma(v_s \cdot v_{ij})\)
- Training objective:
  \[
  \max_{\{v_s\},\{v_{ij}\}} \sum_{\mathcal{F} \in \mathcal{F}} \log([\mathcal{F}])
  \]
- Can we do this for any propositional formulae?
  \[
  [\mathcal{F}] = [r_1(e_i e_j) \land \neg r_2(e_i e_j) \Rightarrow r_3(e_i e_j)]
  \]
Differentiable logic formulae

\[ [F] = \begin{cases} 
\sigma(v_s \cdot v_{ij}) & \text{if } F = r_s(e_i, e_j), \text{ i.e., facts} \\
1 - [A] & \text{if } F = \neg A \\
[A] \ast [B] & \text{if } F = A \land B 
\end{cases} \]

- From negation and conjunction we can build any propositional formula
  - \[ [A \lor B] = 1 - (1 - [A]) \ast (1 - [B]) \]
  - \[ [A \Rightarrow B] = [A]([B] - 1) + 1 \]
- Jointly maximize the log-likelihood of atomic and propositional formulae
  \[ \max_{\{v_s\}, \{v_{ij}\}} \sum_{F \in \mathcal{F}} \log([F]) \]
Backpropagation through structure

(Goller and Küchler, 1996)

\[ r_1(e_i, e_j) \land \neg r_2(e_i, e_j) \Rightarrow r_3(e_i, e_j) \]
Grounding

\[ \forall x, y : r_s(x, y) \Rightarrow r_t(x, y) \]

- Grounding based on all observed facts for premise \( r_s(e_i, e_j) \) and consequent \( r_t(e_i, e_j) \)
- Sample unobserved facts \( r_s(e'_i, e'_j) \) and \( r_t(e'_i, e'_j) \)
- Add propositional formulae to the matrix factorization training objective
  - \[ [r_s(e_i, e_j) \Rightarrow r_t(e_i, e_j)] \]
  - \[ [r_s(e'_i, e'_j) \Rightarrow r_t(e'_i, e'_j)] \]
Experiments

- Relation extraction corpus (Riedel et al., 2013)
  - ~4k textual patterns from New York Times corpus and 151 Freebase relations, ~42k entity-pairs, ~100k training facts
  - Metric: mean average precision (MAP) on manually annotated predictions for Freebase relations

- Zero-shot Relation Learning:
  - All Freebase training facts are removed
  - No alignment between Freebase relations and textual patterns

- Relations with Few Distant Labels:
  - Varying degree of alignment between relations and textual patterns
Evaluation

- Formulae extracted from predictions of a matrix factorization model (Sanchez et al., 2015)
- Annotated manually
- Given these formulae, which method can best make use of them?
  - Logical Inference
  - Post-Factorization Inference
  - Pre-Factorization Inference
  - Joint Optimization
- Example:
  \[ \forall x, y : \#2-minister-\#1(x, y) \Rightarrow \text{person/nationality}(x, y) \]
Zero-shot relation learning
Zero-shot relation learning
Zero-shot relation learning

- MF: 0.03
- Inf: 0.1

weighted Mean Average Precision (wMAP)
Zero-shot relation learning

![Bar graph showing weighted Mean Average Precision (wMAP) for MF, Inf, and Post.]

- MF: 0.03
- Inf: 0.1
- Post: 0.21
Zero-shot relation learning

- MF: 0.03
- Inf: 0.1
- Post: 0.21
- Pre: 0.33
Zero-shot relation learning

![Bar chart showing performance metrics for different methods.]

- MF: 0.03
- Inf: 0.1
- Post: 0.21
- Pre: 0.33
- Joint: 0.38

The chart displays the weighted Mean Average Precision (wMAP) for each method.
Relations with few distant labels

![Graph showing wMAP vs. Fraction of Freebase training facts for different datasets: MF, Joint, Pre, Post, and Inf.](image)
Summary

- Matrix factorization for relation extraction generalizes well
  - Hard to fix mistakes
  - Fails for new or spare relations
- Formalize background knowledge as logical formulae
  - Brittle and no generalization
- Formulae can be injected into embeddings
  - Make logical background knowledge differentiable
  - Jointly optimize over factual and first-order knowledge
  - Learns relations with no or little prior information in databases
  - Generalizes beyond textual patterns mentioned in formulae
- Joint optimization > Pre-factorization inference > Post-factorization inference > Logical inference
Matrix Factorization in Wolfe

```python
def loss(t: Thetas.Term): DoubleTerm = {
  // we sample a positive cell, and memoize the result
  val pos = mem(trainingData.sampleShuffled)
  // based on the memoized positive cell, we sample a negative cell which needs to
  val neg = mem(sampleNegCell(pos))
  // the loss based on positive and negative cell
  log(sigm(score(t)(pos)))) + log(sigm(-score(t)(neg)))) + regularize(t)(pos, neg)
}
```

```python
// the per cell score
def score(theta: Thetas.Term)(cell: Cells.Term): DoubleTerm =
  theta.rows(cell.row) dot theta.cols(cell.col)
```

We are giving a demo at 5pm today!
Thank you!

Questions?

rockt.github.com