Part I
Transfer Learning in Language
Part II
Hal Daumé III
Typical NLP pipeline

The man ate a sandwich

DT NN VB DT NN

Agent Theme

∃ a ∃ t ∃ e man(a) & sandwich(t) & eat(e,a,t) & past(e)

Source Words

Target Words

Analysis

Generation

Morphology
Tagging
Parsing
Role labeling
Interpretation

Source Semantics
Shallowmantics
Syntax
Morphology

Target Semantics
Shallowmantics
Syntax
Morphology

Source Words
Target Words
Pipeline models break down (sorta)

- Tagging + Parsing  
  + 0% / + 3%

- Parsing + Named Entities  
  + 0.5% / + 4%

- Parsing + Role Identification  
  + 0% / - 0.3%
  (upper bound: + 13% )

- Named Entities + Coreference  
  + 0.3% / + 1.3%
  (upper bound: + 8% )

Why? Maybe simpler model already has a lot of the fancier information? Maybe some of these tasks are more related than others?
Example 1: Sentiment analysis for different product types

- In = review
- Out = rating
- Bag of words
- Crawled from Amazon
A probabilistic model for trees

- Kingman’s coalescent is the standard model for the genealogical history of populations.
- It is assumed that each organism has exactly one parent (haploid).
- Thus the genealogy of a population of organisms is a tree.
- Kingman’s coalescent is a particularly elegant and simple distribution over genealogical trees of the population.
From trees to priors...

Place a simple Markov process defined on the tree for $p(X|T)$ which evolves forward in time.
Inference

1. Choose global params:
   \((\mu^{(0)}, \Lambda) \sim \mathcal{NIW}(0, \sigma^2 I, D + 1)\)

2. Choose a tree structure:
   \((\pi, \delta) \sim \text{Coalescent}\)

3. For each non-root \(i \in \pi:\)
   3.1 Choose \(\mu^{(i)} \sim \mathcal{NI}(\mu^{(\pi(i))}, \delta_i \Lambda),\)
      where \(\pi(i)\) is the parent of \(i\)

4. For each domain \(k \in [K]:\)
   4.1 Denote by \(w^{(k)} = \mu^{(i)}\) where \(i\) is the leaf corresponding to \(k\).
   4.2 For each example \(n \in [N_k]:\)
      4.2.1 Choose \(x_n^{(k)} \sim \mathcal{D}(k).\)
      4.2.2 Choose \(y_n^{(k)}\) by \(F(w^{(k)\top} x_n^{(k)})\)
## Experiments (selected)

<table>
<thead>
<tr>
<th>Model</th>
<th>N=100</th>
<th>N=6400</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indp</td>
<td>62.1%</td>
<td>75.8%</td>
</tr>
<tr>
<td>Pool</td>
<td>67.3%</td>
<td>74.5%</td>
</tr>
<tr>
<td>FEDA</td>
<td>63.6%</td>
<td>75.7%</td>
</tr>
<tr>
<td>YaXue</td>
<td>67.8%</td>
<td>72.3%</td>
</tr>
<tr>
<td>Bickel</td>
<td>68.0%</td>
<td>72.5%</td>
</tr>
</tbody>
</table>

**Coal:**
- **Full**: 72.2% | 80.5%
- **Diag**: 71.9% | 80.4%
- **Data**: 70.1% | 75.8%

---

![Graphs showing data distribution](image-url)
Learning task relationships

[Diagram showing the relationship between tasks and a task relationship matrix.]
Task Relationship Learning

- **multitask instance** $\phi_t(x) \in \mathbb{R}^{d(Kd)} = (0, \ldots, 0, x_t, 0, \ldots, 0)_{d(i_t-1)\text{times}}, (0, \ldots, 0)_{d(K-i_t)\text{times}}$

- **compound weight vector** $\mathbf{w}_s^T = (\mathbf{w}_1^T, \mathbf{w}_2^T, \ldots, \mathbf{w}_K^T) \in \mathbb{R}^{Kd}$

- **update rules**: $\mathbf{w}_s = \mathbf{w}_{s-1} + y_t (A \otimes I_d)^{-1} \phi_t$ and $s$ denotes the update number ($s < t$)

  \[
  A = \begin{bmatrix}
  K & -1 & \cdots & -1 \\
  -1 & K & \cdots & -1 \\
  \vdots & \vdots & \ddots & \vdots \\
  -1 & -1 & \cdots & K
  \end{bmatrix}
  \quad \text{and} \quad
  A^{-1} = \frac{1}{K+1} \begin{bmatrix}
  2 & 1 & \cdots & 1 \\
  1 & 2 & \cdots & 1 \\
  \vdots & \vdots & \ddots & \vdots \\
  1 & 1 & \cdots & 2
  \end{bmatrix}
  \]

- **$K \times K$ interaction matrix** $A$ controls the updates

- **update scheme**:
  - **fixed full update** for the current task $i_t$
  - **fixed half update** for the remaining $(K - 1)$ tasks
Joint learning of relationships

- **key idea:** joint minimization of $A$ and $w$
  
  $$\arg\min_{w \in \mathbb{R}^{kd}, A > 0} \left[ D_w(w \| w_s) + D_A(A \| A_s) + \sum_{t} l_t(w) \right]$$

- in this work: hinge loss for $l_t(w)$, mahalanobis distance for $D_w(\cdot \| \cdot)$, log-det divergence and von-neumann divergence for $D_A(\cdot \| \cdot)$

- update rules after **alternating minimization**:
  
  - $w_s = w_{s-1} + y_t(A_{s-1} \otimes I_d)^{-1} \phi_t$
  
  - $A_s = f^{-1}\left(f(A_{s-1}) - \eta \text{ sym} \left(\nabla A \frac{1}{2} \text{tr} (W_{s-1} A W_{s-1}^T)\right)\right)$

  where, $\text{tr}(W_{s-1} A W_{s-1}^T) = w_{s-1}^T (A \otimes I_d) w_{s-1}$, and $W = [w_1, w_2, \ldots, w_K] \in \mathbb{R}^{d \times K}$

- **issue:** when to start updating $A$? our approach:
  
  - initially learn $K$ independent classifiers
  
  - start updating $A$ after reaching **priming duration** (EPOCH)
Experimental Results (sample)

<table>
<thead>
<tr>
<th>Method</th>
<th>20newsgroups</th>
<th>(Standard Deviation) Sentiment</th>
<th>Spam</th>
</tr>
</thead>
<tbody>
<tr>
<td>STL</td>
<td>56.94(±3.32)</td>
<td>66.31(±2.14)</td>
<td>76.45(±1.56)</td>
</tr>
<tr>
<td>IPL</td>
<td>75.20(±2.35)</td>
<td>67.24(±1.40)</td>
<td>91.02(±0.77)</td>
</tr>
<tr>
<td>CMTL</td>
<td>73.14(±2.35)</td>
<td>67.38(±1.82)</td>
<td>90.17(±0.66)</td>
</tr>
<tr>
<td>OMTLLog</td>
<td>81.83(±0.46)</td>
<td>73.49(±0.53)</td>
<td>91.35(±1.12)</td>
</tr>
<tr>
<td>OMTLvon</td>
<td>76.51(±1.54)</td>
<td>67.60(±0.83)</td>
<td>91.05(±1.05)</td>
</tr>
</tbody>
</table>

Accuracy for full training data ($\text{Epoch} = 0.5$).
Transfer Learning

aka: why everything I’ve told you so far isn't useful for some problems...
Domains really are different

- Can you guess what domain each of these sentences is drawn from?

<table>
<thead>
<tr>
<th>Domain</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>News</td>
<td>Many factors contributed to the French and Dutch objections to the proposed EU constitution</td>
</tr>
<tr>
<td>Parliament</td>
<td>Please rise, then, for this minute's silence</td>
</tr>
<tr>
<td>Medical</td>
<td>Latent diabetes mellitus may become manifest during thiazide therapy</td>
</tr>
<tr>
<td>Science</td>
<td>Statistical machine translation is based on sets of text to build a translation model</td>
</tr>
<tr>
<td>Step-mother</td>
<td>I forgot to mention in yesterdays post that I also trimmed an overgrown HUGE hedge that spams the entire length of the front of my house and is about 3' accrossed.</td>
</tr>
</tbody>
</table>
S⁴ ontology of adaptation effects

- **Seen:** Never seen this word before
  - News to medical: “diabetes mellitus”
- **Sense:** Never seen this word used in this way
  - News to technical: “monitor”
- **Score:** The wrong output is scored higher
  - News to medical: “manifest”
- **Search:** Decoding/search erred (ignored)

(inside=old domain outside=new domain)
Translating across domains is hard

<table>
<thead>
<tr>
<th>Old Domain (Parliament)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Original</strong></td>
</tr>
<tr>
<td>monsieur le président, les pêcheurs de homard de la région de l'atlantique sont dans une situation catastrophique.</td>
</tr>
<tr>
<td><strong>Reference</strong></td>
</tr>
<tr>
<td>mr. speaker, lobster fishers in atlantic canada are facing a disaster.</td>
</tr>
<tr>
<td><strong>System</strong></td>
</tr>
<tr>
<td>mr. speaker, the lobster fishers in atlantic canada are in a mess.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>New Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Original</strong></td>
</tr>
<tr>
<td>comprimés pelliculés blancs pour voie orale.</td>
</tr>
<tr>
<td><strong>Reference</strong></td>
</tr>
<tr>
<td>white film-coated tablets for oral use.</td>
</tr>
<tr>
<td><strong>System</strong></td>
</tr>
<tr>
<td>white <em>pelliculés</em> tablets to oral.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>New Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Original</strong></td>
</tr>
<tr>
<td>mode et voie(s) d'administration</td>
</tr>
<tr>
<td><strong>Reference</strong></td>
</tr>
<tr>
<td>method and route(s) of administration</td>
</tr>
<tr>
<td><strong>System</strong></td>
</tr>
<tr>
<td><em>fashion</em> and <em>voie(s)</em> of <em>directors</em></td>
</tr>
</tbody>
</table>

**Key Question:** What went wrong?
Adaptation effects in MT

• Quick observations:
  • New D language model helps (10%-63% improvement)
  • Tuning on new D data helps (10%-90% improvement)
  • Weighting new D data helps (4%-150% improvement)

• Identifying errors in MT (w/o parallel newD data):
  • **Seen**: old-only model + unseen input word pairs
  • **Sense**: old-only model + seen input/unseen output pairs
  • **Score**: intersect old and mixed model, score from old

<table>
<thead>
<tr>
<th></th>
<th>News</th>
<th>Medical</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Seen</strong></td>
<td>Little effect</td>
<td>~ 40% of error</td>
</tr>
<tr>
<td><strong>Sense</strong></td>
<td>Little effect</td>
<td>~ 40% of error</td>
</tr>
<tr>
<td><strong>Score</strong></td>
<td>~ 90% of error</td>
<td>~ 20% of error</td>
</tr>
</tbody>
</table>

(As measured by Bleu score)

Consistent in:
* movie subtitles
* scientific pubs
* PHP tech docs
Translating across domains is hard

<table>
<thead>
<tr>
<th>Dom</th>
<th>Most frequent OOV Words</th>
</tr>
</thead>
<tbody>
<tr>
<td>News</td>
<td>behavior, favor, neighbors, fueled</td>
</tr>
<tr>
<td>(17%)</td>
<td>neighboring, abe, zhao, phelps</td>
</tr>
<tr>
<td>Medical</td>
<td>renal, hepatic, subcutaneous, irbesartan</td>
</tr>
<tr>
<td>(49%)</td>
<td>ribavirin, olanzapine, serum</td>
</tr>
<tr>
<td></td>
<td>dl, eine, hydrochlorothiazide</td>
</tr>
<tr>
<td></td>
<td>subcutaneous, patienten</td>
</tr>
<tr>
<td></td>
<td>ritonavir, erythropoietin</td>
</tr>
<tr>
<td></td>
<td>pharmacokinetics, efavirenz</td>
</tr>
<tr>
<td>Movies</td>
<td>gonna, yeah, mom, hi</td>
</tr>
<tr>
<td>(44%)</td>
<td>b****, daddy, s***</td>
</tr>
<tr>
<td></td>
<td>f*****g, gotta, bye</td>
</tr>
<tr>
<td></td>
<td>uh, namely, bye</td>
</tr>
</tbody>
</table>

[Daumé III & Jagarlamudi, 2011]
Dictionary mining for “seen” errors

- Find frequent terms in new domain
- Use those that exist in old domain as “training data”
- Extract context and orthographic features
- Find low-dimensional subspace on training data (CCA)

- Pair input words with <=5 output words
- Add four features to SMT model
- Rerun parameter tuning

![Diagram showing the process](link-to-diagram)

(Bleu score improvements)

<table>
<thead>
<tr>
<th>DE</th>
<th>FR</th>
</tr>
</thead>
<tbody>
<tr>
<td>News</td>
<td>+0.80</td>
</tr>
<tr>
<td>Emea</td>
<td>+1.44</td>
</tr>
<tr>
<td>Subs</td>
<td>+0.13</td>
</tr>
<tr>
<td>PHP</td>
<td>+0.28</td>
</tr>
</tbody>
</table>
Senses are domain/language specific

French
- courir
- éxécuter

English
- run
- virus
- window

Japanese
- 走る
- 病原体
- 窓
Automatically identifying new senses

- Context + existence of translations in comparable data

is a window of opportunity
have a window of opportunity
in the run up to
we run the risk

via une fenêtre insérée.
vers ma fenêtre ou vers
voulons pas courir le risque
, sans courir le risque

dans la fenêtre. cet
dans la fenêtre. </s>
courir not found

the browser window's
in the window to give
time to run when applied
or have run vcvars.bat,

ne pouvez exécuter que les
pour l' exécuter elle va
Spotting New Senses

- **Binary classification problem:**
  - **+ve:** French token has previously unseen sense
  - **-ve:** French token is used in a known way

- **Lots of features considered...**
  - Frequency of words/translations in each domain
  - Language model perplexities across domains
  - Topic model “mismatches”
  - Marginal matching features
  - Translation “flow” impedence

---

**Given:**
- A joint $p(x,y)$ in the old domain
- Marginals $q(x)$ and $q(y)$ in the new domain

**Recover:**
- Joint $q(x,y)$ in the new domain

We formulate as a L1-regularized linear program.

**Easier alternative:** we have many such $q(x)$ and $q(y)$s
Experimental Results

- **EMEA**
  - One Feature
  - Two Features
  - Three Features
  - All Features

- **Science**
  - One Feature
  - Two Features
  - Three Features
  - All Features

- **Subs**
  - One Feature
  - Two Features
  - Three Features
  - All Features

Selected features:
- **EMEA**: ppl || matchm flow || matchm topics flow
- **Science**: ppl || matchm ppl || matchm topics ppl
- **Subs**: topics || matchm topics || matchm topics flow
Conclusions

- **Transfer Learning...**
  - Assuming fixed task/domain relatedness is a bad idea
  - Key question: what type of representation is “right”?
  - Can do subspaces, trees, clusters, etc. etc. etc.

- **In Language...**
  - ML addresses only part of the adaptation picture
  - So far, specialized approaches for addressing other parts
    - Mining translations from comparable data
    - Automatically spotting new word senses

Thanks! Questions?