CS11-747 Neural Networks for NLP

Document Level Models

Zhengzhong Liu (Hector)

Site
https://phontron.com/class/nn4nlp2017/
NN and some NLP tasks

Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, 'and what is the use of a book,' thought Alice 'without pictures or conversation?'

\[
P(w_{1}= \text{of} \mid w_{2}= \text{tired}) = 1 \\
P(w_{1}= \text{of} \mid w_{2}= \text{use}) = 1 \\
P(w_{1}= \text{sister} \mid w_{2}= \text{her}) = 1 \\
P(w_{1}= \text{beginning} \mid w_{2}= \text{was}) = 1/2 \\
P(w_{1}= \text{reading} \mid w_{2}= \text{was}) = 1/2
\]

Language Models

Parsing

Germany's representative to the European Union's veterinary committee Werner Zwingman said on Wednesday consumers should ...

Classification

Entity Tagging
Their Counter-part in Documents

<table>
<thead>
<tr>
<th>Entity</th>
<th>Sentence</th>
<th>Document</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity Tagging</td>
<td>Coreference</td>
<td></td>
</tr>
<tr>
<td>Semantic Parsing; Syntactic Parsing</td>
<td>Discourse Parsing</td>
<td></td>
</tr>
<tr>
<td>Word Prediction</td>
<td>Sentence/Discourse Element Prediction</td>
<td></td>
</tr>
</tbody>
</table>

Language Model

<table>
<thead>
<tr>
<th>Classification</th>
<th>Sentence Classification</th>
<th>Document Classification</th>
</tr>
</thead>
</table>

70% 15% 15%

Recovers structures of documents
Document Problems: Entity Coreference

Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. A renowned speech therapist was summoned to help the King overcome his speech impediment...

Example from Ng, 2016

• Step 1: Identify Noun Phrases mentioning an entity (note the difference from named entity recognition).

• Step 2: Cluster noun phrases (mentions) referring to the same underlying world entity.
One may think coreference is simply a clustering problem of given Noun Phrases.

Detecting relevant noun phrases is a difficult and important step.

Knowing the correct noun phrases affect the result a lot.

Normally done as a preprocessing step.
Components of a Coreference Model

• Like a traditional machine learning model:

  • We need to know the instances (e.g. shift-reduce operations in parsing).

  • We need to design the features.

  • We need to optimize towards the evaluation metrics.

  • Search algorithm for structure (covered in later lectures).
Coreference Models: Instances

• Coreference is a structured prediction problem:
  • Possible cluster structures are in exponential number of the number of mentions. (Number of partitions)
  • Models are designed to approximate/explore the space, the core difference is the way each instance is constructed:
    • Mention-Pair Model
    • Entity-Mention Model
    • Mention-Ranking Model
    • Latent Tree Models
  • Mimic the cluster creation process of human.

Which mention to link to?
Mention Pair Models

- The simplest one: Mention Pair Model:
  - Classify the coreference relation between every 2 mentions.

- Simple but many drawbacks:
  - May result in conflicts in transitivity.
  - Too many negative training instances.
  - Do not capture entity/cluster level features.
  - No ranking of instances.

- Examples:
  - Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch.
  - A renowned speech therapist was summoned to help the King overcome his speech impediment...

- Correct matches:
  - ✓: Queen Elizabeth <-> her
  - ✓: Queen Elizabeth <-> her

- Incorrect matches:
  - ❌: Queen Elizabeth <-> husband
  - ❌: Queen Elizabeth <-> King George VI
  - ❌: Queen Elizabeth <-> a viable monarch

...
Entity Models: Entity-Mention Models

- Entity-Mention Models
  - Create an instance between a mention and a previous* cluster.

Daume & Marcu (2005); Cullotta et al. (2007)

Example Cluster Level Features:
- Are the genders all compatible?
- Is the cluster containing pronouns only?
- Most of the entities are the same gender?????
- Size of the clusters?

Problems:
- No ranking between the antecedents.
- Cluster level features are difficult to design.

* This process often follows the natural discourse order, so we can refer to partial build clusters.
Entity Models: Entity-Centric Models
Clark and Manning (2015)

• Entity Centric Models
  • Create an instance between two clusters.
  • Allow building a entity representation.

Problems:
• Cluster level features are difficult to design. (recurring problem)
• No direct guidance of entity creation process

Learning Algorithm
• Build up clusters during learning (normally agglomerative)
• No cluster creation gold standard!!
  • “Create” gold standard to guide the clusters.
• Train with RL: Clark and Manning (2015) trained it with DAgger.
Ranking Models

- Added relative importance to antecedents.
  - Easy-first intuition, some decisions are easier than the others.
  - Help deal with imbalance between positive and negative.
  - Anaphora problem: what if a mention does not have an antecedent? (Create a NULL mention)
- Mention Ranking (Currently more popular)
  - Ranking previous mentions. (Durrett & Klein 2013, Ma et.al 2016)
- Entity Ranking
  - Rank preceding clusters, not individual mentions. (Rahman & Ng, 2009)
Ranking Model: Mention Ranking
(Durrett and Klein, 2013)

[ Voters]₁ agree when [they]₁ are given a [chance]₂ to decide if [they]₁ ...

A Log-Linear probabilistic Model
• Create a antecedent structure (a₁, a₂, a₃, a₄): where each mention need to decide a ranking of the antecedents
• Problem: No Gold Standard antecedent structure?
  • Sum over all possible structures licensed by the gold cluster
Ranking Model:
Entity Ranking
(Rahman & Ng, 2009)

Rank previous clusters for a given mention. Similarly, a NULL cluster is added to the antecedents. Rahman & Ng use a complex set of features (39 feature templates)
Latent Tree Models

(Bjorkelund and Kuhn, 2014)

Latent Tree Model share some similarities with the mention ranking models.

Each subtree under the root represent a cluster.

Trained as **structured perceptron**
- Create a antecedent structure (as a tree), where each mention need to decide which antecedent to linked to (similar to a ranking)
- Problem: No Gold Standard antecedent tree? (Hence called the Latent Tree)
  - **Pick the highest scored tree structure within** all possible structures licensed by the gold cluster
What's the role of Neural Networks here?
Problems in Coreference: revisited

- **Instance Problem**
  - We’ve introduced 4 different modeling methods, many seem to work in their own settings.

- **Feature Problem**
  - The core of the success may still be the feature problem. For example, Bjorkelund and Kuhn use a decision tree for feature induction. Durrett and Klein conduct careful feature engineering and selection.

- **Metric Problem**: clustering metric is (very) difficult to compute (any thoughts?)
• Five types of operation to transform coreference decisions.
• The combination of the operations creates 7 types of errors.
Error Driven Analysis
(Kummerfeld and Klein, 2013)

- Five types of operation to transform coreference decisions.
- The combination of the operations creates 7 types of errors.

<table>
<thead>
<tr>
<th>System</th>
<th>Metric F-Scores</th>
<th>Span Error</th>
<th>Conflated Entities</th>
<th>Extra Mention</th>
<th>Extra Entity</th>
<th>Divided Entity</th>
<th>Missing Mention</th>
<th>Missing Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mention</td>
<td>MUC</td>
<td>B³</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BERKELEY</td>
<td>75.57</td>
<td>66.43</td>
<td>66.17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IMS</td>
<td>72.96</td>
<td>64.71</td>
<td>64.73</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STANFORD-T</td>
<td>71.21</td>
<td>61.40</td>
<td>63.06</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STANFORD</td>
<td>58.56</td>
<td>48.37</td>
<td>56.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RECONCILE</td>
<td>46.45</td>
<td>49.40</td>
<td>54.90</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BART</td>
<td>56.61</td>
<td>46.00</td>
<td>52.56</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UIUC</td>
<td>50.60</td>
<td>45.21</td>
<td>52.88</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHERRY_PICKER</td>
<td>41.10</td>
<td>40.71</td>
<td>51.39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Easy Victories & Uphill Battles

• A mention ranking model (We’ve actually covered its model in previous slides).

• Error type based loss in cost function:

  • Trained with softmax-margin cost (a way to add cost sensitive training to log-linear models).

  • Combined loss: \( l(a, C^*) = a_{FA} FA(a, C^*) + a_{FN} FN(a, C^*) + a_{WL} WL(a, C^*) \)

  • FA (False Anaphora), FN (False New), WL (Wrong Link)
Easy Victories & Uphill Battles
(Durrett and Klein, 2013)

• Easy Victories from Surface (lexical) Features:
  • Ignore all many complex features, all replaced with surface features.
  • **Data driven features** beat Heuristic driven (Sounds familiar?).
  • Many heuristic features can be captured (implicitly) by surface features:
    • Number, gender, person can be encoded in pronouns.
    • Centering theory: verb before or after can indicate subj, obj.
    • Definiteness: first word of a mention will encode that.
## Easy Victories & Uphill Battles

<table>
<thead>
<tr>
<th>Feature name</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Features on the current mention</td>
<td></td>
</tr>
<tr>
<td>[ANAPHORIC] + [HEAD WORD]</td>
<td>41371</td>
</tr>
<tr>
<td>[ANAPHORIC] + [FIRST WORD]</td>
<td>18991</td>
</tr>
<tr>
<td>[ANAPHORIC] + [LAST WORD]</td>
<td>19184</td>
</tr>
<tr>
<td>[ANAPHORIC] + [PRECEDING WORD]</td>
<td>54605</td>
</tr>
<tr>
<td>[ANAPHORIC] + [FOLLOWING WORD]</td>
<td>57239</td>
</tr>
<tr>
<td>[ANAPHORIC] + [LENGTH]</td>
<td>4304</td>
</tr>
<tr>
<td>Features on the antecedent</td>
<td></td>
</tr>
<tr>
<td>[ANTECEDENT HEAD WORD]</td>
<td>57383</td>
</tr>
<tr>
<td>[ANTECEDENT FIRST WORD]</td>
<td>24239</td>
</tr>
<tr>
<td>[ANTECEDENT LAST WORD]</td>
<td>23819</td>
</tr>
<tr>
<td>[ANTECEDENT PRECEDING WORD]</td>
<td>53421</td>
</tr>
<tr>
<td>[ANTECEDENT FOLLOWING WORD]</td>
<td>55718</td>
</tr>
<tr>
<td>[ANTECEDENT LENGTH]</td>
<td>4620</td>
</tr>
<tr>
<td>Features on the pair</td>
<td></td>
</tr>
<tr>
<td>[EXACT STRING MATCH (T/F)]</td>
<td>47</td>
</tr>
<tr>
<td>[HEAD MATCH (T/F)]</td>
<td>46</td>
</tr>
<tr>
<td>[SENTENCE DISTANCE, CAPPED AT 10]</td>
<td>2037</td>
</tr>
<tr>
<td>[MENTION DISTANCE, CAPPED AT 10]</td>
<td>1680</td>
</tr>
</tbody>
</table>

### Final Feature Set

<table>
<thead>
<tr>
<th>Feature name</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Features of the SURFACE system</td>
<td>418704</td>
</tr>
<tr>
<td>Features on the current mention</td>
<td></td>
</tr>
<tr>
<td>[ANAPHORIC] + [CURRENT ANCESTRY]</td>
<td>46047</td>
</tr>
<tr>
<td>Features on the antecedent</td>
<td></td>
</tr>
<tr>
<td>[ANTECEDENT ANCESTRY]</td>
<td>53874</td>
</tr>
<tr>
<td>[ANTECEDENT GENDER]</td>
<td>338</td>
</tr>
<tr>
<td>[ANTECEDENT NUMBER]</td>
<td>290</td>
</tr>
<tr>
<td>Features on the pair</td>
<td></td>
</tr>
<tr>
<td>[HEAD CONTAINED (T/F)]</td>
<td>136</td>
</tr>
<tr>
<td>[EXACT STRING CONTAINED (T/F)]</td>
<td>133</td>
</tr>
<tr>
<td>[NESTED (T/F)]</td>
<td>355</td>
</tr>
<tr>
<td>[DOC TYPE] + [SAME SPEAKER (T/F)]</td>
<td>437</td>
</tr>
<tr>
<td>[CURRENT ANCESTRY] + [ANT. ANCESTRY]</td>
<td>2555359</td>
</tr>
</tbody>
</table>
Some Possible Improvements w/ NN

• Train towards the metric using Deep RL.

• Learn the features with embeddings since most of them can be captured by surface features.

• Can some features be captured better with NN?

• Train the full system to reduce specific error types:
  • which errors specifically?
Coreference Resolution w/ Entity-Level Distributed Representations
Clark & Manning (2015)

- Mention Pair Model and Cluster Pair model to capture representation
- Typical Coreference Features are used as embeddings or on-hot features
- Mention Pair Features are fed to the cluster pair features, followed by pooling
- Heuristic Max-Margin as in Wiseman et al. (2015) and Durrett & Klein (2013)
- Cluster merging as with Policy Network (MERGE or PASS)
- Trained with SEARN (Daume III et al., 2009)
Deep Reinforcement Learning for Mention-Ranking Coreference Models
Clark & Manning (2016)

• A continuous of the previous model:
  • Same features and structure.

• Objective changed: reinforcement learning
  • Choosing which previous antecedent is considered as an action of the agent.
  • The final reward is one of the 4 main evaluation metric in coreference (B-Cubed).
  • Best model is reward-rescaled reinforcement method.
Cluster Features w/ Neural Network
Wiseman et.al (2016)

- Cluster level features are difficult to capture.
- Example cluster level features:
  - most-female=true (how to define most?).
  - Pronoun sequence: C-P-P = true.
- Use RNN to embed features from multiple mentions into a single representation.
  - No hand designed cluster level feature templates.
End-to-End Neural Coreference
Lee et.al (2017)

• 2 main contributions by this paper:

  • Can we represent all features with a more typical neural network embedding way?

  • Can neural network allow errors to flow end-to-end? All the way to mention detection?

  • This solves another type of error (span error), which is not previously handled.
End-to-End Neural Coreference (Span Model)

- Build mention representation from word representation (all possible spans)
- Head extracted by self-attention.
End-to-End Neural Coreference (Coreference Model)

- Coreference model is similar to a mention ranking.
- Coreference score consist of multiple scores.
- Simple max-likelihood (not the cost sensitive method by Durrett, why?)
Quality of Mentions

- Build mention representation from word representation (all possible spans)
- Head extracted by self-attention.
Ablations of modules

<table>
<thead>
<tr>
<th></th>
<th>Avg. F1</th>
<th>Δ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our model (ensemble)</td>
<td>69.0</td>
<td>+1.3</td>
</tr>
<tr>
<td>Our model (single)</td>
<td>67.7</td>
<td></td>
</tr>
<tr>
<td>distance and width features</td>
<td>63.9</td>
<td>-3.8</td>
</tr>
<tr>
<td>GloVe embeddings</td>
<td>65.3</td>
<td>-2.4</td>
</tr>
<tr>
<td>speaker and genre metadata</td>
<td>66.3</td>
<td>-1.4</td>
</tr>
<tr>
<td>head-finding attention</td>
<td>66.4</td>
<td>-1.3</td>
</tr>
<tr>
<td>character CNN</td>
<td>66.8</td>
<td>-0.9</td>
</tr>
<tr>
<td>Turian embeddings</td>
<td>66.9</td>
<td>-0.8</td>
</tr>
</tbody>
</table>

Table 2: Comparisons of our single model on the development data. The 5-model ensemble provides a 1.3 F1 improvement. The head-finding attention, features, and all word representations contribute significantly to the full model.
Error Type Revisited

1. Alter Span
2. Split
3. Remove
4. Introduce
5. Merge

<table>
<thead>
<tr>
<th>System</th>
<th>Metric F-Scores</th>
<th>Span Error</th>
<th>Conflated Entities</th>
<th>Extra Mention</th>
<th>Extra Entity</th>
<th>Divided Entity</th>
<th>Missing Mention</th>
<th>Missing Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mention</td>
<td>MUC</td>
<td>B³</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BERKELEY</td>
<td>75.57</td>
<td>66.43</td>
<td>66.17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IMS</td>
<td>72.96</td>
<td>64.71</td>
<td>64.73</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STANFORD-T</td>
<td>71.21</td>
<td>61.40</td>
<td>63.06</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STANFORD</td>
<td>58.56</td>
<td>48.37</td>
<td>56.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RECONCILE</td>
<td>46.45</td>
<td>49.40</td>
<td>54.90</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BART</td>
<td>56.61</td>
<td>46.00</td>
<td>52.56</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UIUC</td>
<td>50.60</td>
<td>45.21</td>
<td>52.88</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHERRYPicker</td>
<td>41.10</td>
<td>40.71</td>
<td>51.39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Kummerfeld and Klein, 2013
The projections are in the neighborhood of 50 cents a share to 75 cents, compared with a restated $1.65 a share a year earlier, when profit was $107.8 million on sales of $435.5 million.
Document Problems: Discourse Parsing

- Parse a piece of text into relations between discourse units (EDUs).
- Researchers mainly used the Rhetorical Structure Theory (RST) formalism, which forms a tree of relations.
Recursive Deep Models for Discourse Parsing

• Recursive NN for discourse parsing (similar to Socher’s recursive parsing)
• First determine whether two spans should be merged (Binary)
• Then determine the relation type

Li et.al (2014)
Discourse Parsing w/ Attention-based Hierarchical Neural Networks

Li et.al (2016)

- Hierarchical bi-LSTM to learn composition scoring.
- Augmented with attention mechanism. (Span is long)
- 2 Bi-LSTMs: first used to capture the representation of an EDU, then combine EDU representation into larger representation
- CKY Parsing
Discourse Structure can help represent documents

Ji and Smith (2017)

- This work shows that the document representation can be built with discourse structure.
- Similar to a representation of sentence using recursive NN on parse tree.
- They reported better sentiment analysis and document topic classification.
Implicit Discourse Connection
Classification w/ Adversarial Objective
(Qin et al. 2017)

- Idea: implicit discourse relations are not explicitly marked, but would like to detect them if they are

- Text with explicit discourse connectives should be the same as text without!
## Discourse Prediction

**Context**

<table>
<thead>
<tr>
<th>Discourse</th>
<th>Right Ending</th>
<th>Wrong Ending</th>
</tr>
</thead>
<tbody>
<tr>
<td>Karan was assigned a roommate her first year of college. Her roommate asked her to go to a nearby city for a concert. Karen agreed happily. The show was absolutely exhilarating.</td>
<td>Karen became good friends with her roommate.</td>
<td>Karen hated her roommate.</td>
</tr>
<tr>
<td>Jim got his first credit card in college. He didn’t have a job so he bought everything on his card. After he graduated he amounted a $10,000 debt. Jim realized that he was foolish to spend so much money.</td>
<td>Jim decided to devise a plan for repayment.</td>
<td>Jim decided to open another credit card.</td>
</tr>
<tr>
<td>Gina misplaced her phone at her grandparents. It wasn’t anywhere in the living room. She realized she was in the car before. She grabbed her dad’s keys and ran outside.</td>
<td>She found her phone in the car.</td>
<td>She didn’t want her phone anymore.</td>
</tr>
</tbody>
</table>

**The Test**

Jennifer has a big exam tomorrow. She got so stressed, she pulled an all-nighter. She went into class the next day, weary as can be. Her teacher stated that the test is postponed for next week. Jennifer felt bittersweet about it.

**The Hurricane**

Morgan and her family lived in Florida. They heard a hurricane was coming. They decided to evacuate to a relative’s house. They arrived and learned from the news that it was a terrible storm. They felt lucky they had evacuated when they did.

**Spaghetti Sauce**

Tina made spaghetti for her boyfriend. It took a lot of work, but she was very proud. Her boyfriend ate the whole plate and said it was good. Tina tried it herself, and realized it was disgusting. She was touched that he pretended it was good to spare her feelings.
Document Problems: Discourse Unit Prediction

Referent Prediction
Corpus from (Modi et.al. 2017)

ROS Story corpus (Mostafazade et.al. 2017)

Predicting the next entity/sentence given previous sentences
Predicting Discourse Units are similar to Language Modeling

- Pichotta and Mooney, 2016 use RNN to predict the next event.
  - Basically Sentence-Level Language Models (of events)
- Peng and Roth, 2016 introduced Semantic Language Model
  - *Kevin was robbed by Robert, but the police mistakenly arrested him.*
  - Frame sequence: [f1, dis1, f2, dis2, …]
  - Entity sequence: [e1, dis1, e2, dis2, …]
  - Applied to coreference resolution and shallow discourse parsing.
Story Completion Task

<table>
<thead>
<tr>
<th>Context</th>
<th>Incorrect Ending</th>
<th>Correct Ending</th>
</tr>
</thead>
<tbody>
<tr>
<td>He didn’t know how the television worked. He tried to fix it, anyway. He climbed up on the roof and fiddled with the antenna. His foot <strong>slipped</strong> on the wet shingles and he went tumbling down.</td>
<td>He decided that was fun and to try tumbling again.</td>
<td>Thankfully, he <strong>recovered</strong>.</td>
</tr>
<tr>
<td>Pam thought her front yard looked boring. So she decided to buy several plants. And she placed them in her front yard. She was <strong>proud</strong> of her work.</td>
<td>Pam was <strong>upset</strong> at herself.</td>
<td>Pam was <strong>satisfied</strong>.</td>
</tr>
<tr>
<td>Maria smelled the fresh Autumn air and decided to celebrate. She wanted to make candy apples. She picked up the ingredients at a local market and headed home. She cooked the candy and prepared the apples.</td>
<td>Maria’s apple <strong>pie</strong> was delicious.</td>
<td>She enjoyed the candy apples.</td>
</tr>
</tbody>
</table>

- Snigdha et.al. (2017) use the Semantic LM learnt by Peng et.al. (2016) as a feature to learn next sentence.
- Cai et.al (2017) use LSTM to encode words as sentences, then encode a series of sentences, to predict next sentence.
Why Discourse LM?

• A normal language model can help predict the next word, very useful in speech recognition, translation, etc.

• A discourse language model help predict the next entity/event, potentially useful for:
  • Information extraction.
  • Entity Coreference (Hey, we just talked about it! Let’s elaborate!).
Solving Hard Coreference with LM
(The uphill battle!)
Peng et. al. (2015)

• The older students were bullying the younger ones, so we [rescued/punished] **them**.

• Robert was robbed by Kevin, and **he** is [arrested/rescued] by police.

• Semantic LMs are useful for solving difficult coreference problems.
  
  • They capture common senses that are not accessible in surface features.

• Peng et. al. (2015, 2016) shows performance improvement of these cases.
Questions?