Announcements

• No lecture on **Dec 12th**.

• All the project meetings scheduled on **Dec 10th**
  • Submit checkpoint reports by Dec 9th (optional but recommended!)
  • Sign up project meetings!
Overview

- What is coreference resolution?
- Four types of coreference models
- End-to-end coreference resolution
- Evaluation and results
Discourse

- Text is more than the sum of its individual sentences/utterances.
- Discourse processing: NLP beyond the sentence/utterance boundary.
  - Monologue
  - Dialogue (multi-party)
- Topics:
  - Topic segmentation
  - **Coreference resolution**
  - Coherence
  - Discourse relations
  - ...


Coreference resolution

• Identify all mentions that refer to the same real world entity

Barack Obama nominated Hillary Rodham Clinton as his secretary of state on Monday. He chose her because she had foreign affairs experience as a former First Lady.
Coreference resolution

- Identify all **mentions** that refer to the same real world entity

```
Barack Obama nominated Hillary Rodham Clinton as his secretary of state on Monday. He chose her because she had foreign affairs experience as a former First Lady.
```
Coreference resolution

- Identify all mentions that refer to the same real world entity

Barack Obama nominated Hillary Rodham Clinton as his secretary of state on Monday. He chose her because she had foreign affairs experience as a former First Lady.
Coreference resolution

- Identify all mentions that refer to the same real world entity

Barack Obama nominated Hillary Rodham Clinton as his secretary of state on Monday. He chose her because she had foreign affairs experience as a former First Lady.
Coreference resolution

- Identify all **mentions** that refer to the same real world entity

**Input**

Barack Obama nominated Hillary Rodham Clinton as his secretary of state on Monday. He chose her because she had foreign affairs experience as a former First Lady.

**Output**

- \{Barack Obama, his, He\}
  
- \{Hillary Rodham Clinton, secretary of state, her, she, First Lady\}
Try coref systems yourself!

- https://corenlp.run
Try coref systems yourself!

- https://demo.allennlp.org/coreference-resolution/

Document

Paul Allen was born on January 21, 1953, in Seattle, Washington, to Kenneth Sam Allen and Edna Faye Allen. Allen attended Lakeside School, a private school in Seattle, where he befriended Bill Gates, two years younger, with whom he shared an enthusiasm for computers. Paul and Bill used a teletype terminal at their high school, Lakeside, to develop their programming skills on several time-sharing computer systems.

Document

Barack Obama nominated Hillary Rodham Clinton as his secretary of state on Monday. He chose her because she had foreign affairs experience as a former First Lady.
Applications

• Full text understanding
• Information extraction, question answering, summarization..

Barack Obama
From Wikipedia, the free encyclopedia

Obama was born in Honolulu, Hawaii. After graduating from Columbia University in 1983, he worked as a community organizer in Chicago. In 1988, he enrolled in Harvard Law School, where he was the first black president of the Harvard Law Review. After graduating, he became a civil rights attorney and an academic, teaching constitutional law at the University of Chicago Law School from 1992 to 2004.

• he = Barack Obama
• (Barack Obama, schools_attended, Harvard University)
Applications

• Languages have different features for gender, number, dropped pronouns, etc.

• “I love knowledge,” he said.
• “I love knowledge,” she said.
Coreference vs Entity Linking

Donald Layton_1 took the helm of Freddie Mac_2 after his_1 ...

en.wikipedia.org/wiki/Freddie_Mac

(Image credit: Greg Durrett)
Winograd Schema problems

- Some cases of coreference require world knowledge or common sense reasoning to solve.

They city council denied the demonstrators a permit because they feared violence.

They city council denied the demonstrators a permit because they advocated violence.

The trophy didn’t fit into the suitcase because it was too large.

The trophy didn’t fit into the suitcase because it was too small.
Winograd Schema problems

The Winograd Schema Challenge: recently proposed as an alternative to the Turing test

- (Levesque 2013): “On our best behavior”
- If you’ve fully solved coreference, arguably you’ve solved AI

http://commonsensereasoning.org/winograd.html
Coreference: some linguistics

- **Coreference** is when two mentions refer to the same entity in the world
  - *Barack Obama* traveled to.. *Obama*...

- A related linguistic concept is **anaphora**: when a term (anaphor) refers to another term (antecedent)
  - The interpretation of the anaphor is in some way determined by the interpretation of the antecedent
  - *Barack Obama* said *he* would sign the bill.
Coreference: some linguistics

- Coreference with named entities

  text

  Barack Obama

  Obama

  world

- Anaphora

  text

  Barack Obama

  he

  world
Coreference: some linguistics

• Not all anaphoric relations are coreferential

bridging anaphora

We went to see a concert last night. The tickets were really expensive.
Coreference: some linguistics

Anaphora vs cataphora

- Usually the antecedent comes before the anaphor (e.g., a pronoun) but not always
- Cataphora: mentioned before the their referents

Even before she saw it, Dorothy had been thinking about the Emerald City every day.

More linguistic background in J & M: Chapter 22.1
Coreference resolution in two steps

• Mention detection (easy)

“[I] voted for [Nader] because [he] was most aligned with [[my] values],” [she] said

  Mentions can be nested!

• Mention clustering (hard)

  “[I] voted for [Nader] because [he] was most aligned with [[my] values],” [she] said
Mention detection

- Mention: span of text referring to some entity
- Three kinds of mentions:
  - #1: Pronouns
    - I, your, it, she, him
  - #2: Named entities
    - “Barack Obama” “Princeton University”
  - #3: Noun phrases
    - “a dog”, “the big fluffy cat stuck in the tree”

Part-of-speech tagger

named entity recognizer

constituency parer
Mention detection: not so simple

- Making all pronouns, named entities, and noun phrases as mentions over-generates mentions

- Are these mentions?
  - It has been ten years
  - Every student
  - No student
  - 100 miles
Mention detection: not so simple

- We can train a classifier to filter out spurious mentions

- Much more common: keep all mentions as “candidate mentions”
  - After your coreference system is done, discard all singleton mentions (ones that have not been marked as coreference with anything else)

- Recent work: it is possibly to do mention detection and clustering end-to-end instead of in two steps
  - We will cover this later!

(Lee et al, 2017): End-to-end Neural Coreference Resolution
Four kinds of coreference models

- Rule-based (pronominal anaphora resolution)
- Machine learning models:
  - Mention pair
  - Mention ranking
  - Clustering-based

“I voted for Nader because he was most aligned with my values,” she said.

Coreference Cluster 1
Coreference Cluster 2
Hobbs algorithm (1978)

- An algorithm that searches parse trees for antecedents of a pronoun
  - Starting at the NP node immediately dominating the pronoun
  - Search previous trees, in order of recency, left-to-right, breadth-first
  - Looking for the first match of the correct gender and number (male vs female, singular vs plural)

- 72.7% accuracy based on the 100 examples he analyzed
Hobbs algorithm (1978)

The castle in Camelot remained the residence of the king until he moved it to London.
Mention-pair models

- Train a binary classifier that assigns every pair of mentions a probability of being coreferent: $p(m_i, m_j)$

“I voted for Nader because he was most aligned with my values,” she said.

Coreference Cluster 1
Coreference Cluster 2

coreferent with she?
Mention-pair models

"I voted for Nader because he was most aligned with my values," she said.

Positive examples: want $p(m_i, m_j)$ to be near 1

Negative examples: want $p(m_i, m_j)$ to be near 0

$$J = - \sum_{i=2}^{N} \sum_{j=1}^{i} y_{ij} \log p(m_j, m_i)$$

Iterate through candidate antecedents (previously occurring mentions)
Coreferent mentions pairs should get high probability, others should get low probability

Iterate through mentions
Mention-pair models

Testing time:
- Run your own mention detector
- Pick some threshold (e.g., 0.5) and add coreference links
- Take the transitive closure to get the clustering

*Diagram showing coreference links between mentions.*
**Mention-pair models**

**Disadvantage:** suppose we have a long document with the following mentions: Ralph Nader... he... his.. him.. <several paragraphs>... voted for Nader because he..

Many mentions only have one clear antecedent but we are asking the model to predict all of them.

**Solution:** instead train the model to predict only one antecedent for each mention (more linguistically plausible)
Mention-ranking models

- Assign each mention its highest scoring candidate antecedent according to the model
- Add a dummy NA mention to decline linking the current mention to anything ("singleton" or "first" mention)

![Diagram showing mention-ranking models](image)

\[
p(\text{NA, she}) = 0.1 \\
p(\text{l, she}) = 0.5 \\
p(\text{Nader, she}) = 0.1 \\
p(\text{he, she}) = 0.1 \\
p(\text{my, she}) = 0.2
\]

Apply a softmax over the scores for candidate antecedents so probabilities sum to 1
Mention-ranking models

- Training time: only clustering information is observed (no annotation of “antecedent”), so we optimize the marginal log-likelihood of all the correct antecedents.

\[
J = \sum_{i=2}^{N} - \log \left( \sum_{j=1}^{i-1} \mathbb{1}(y_{ij} = 1)p(m_j, m_i) \right)
\]

Iterate over all the mentions in the document
Usual trick of taking negative log to go from likelihood to loss

- Testing time: same as mention-pair but we only pick one antecedent for each mention
Clustering-based models

- Coreference is a clustering task, so let’s use a clustering algorithm directly.
- Start with each mention in its own singleton cluster
- Merge a pair of clusters at each step and use a model to score which cluster merges are good.
Clustering-based models

Google recently ... the company announced Google Plus ... the product features ...

Cluster 1: Google

Cluster 2: the company

Cluster 3: Google Plus

Cluster 4: the product

$s(c_1, c_2) = 5$  ✔ merge

Cluster 1: Google

Cluster 2: the company

Cluster 3: Google Plus

$s(c_2, c_3) = 4$  ✔ merge

Cluster 1: Google

Cluster 2: the company

Cluster 2: Google Plus

$s(c_1, c_2) = -3$  ✗ do not merge
Clustering-based models

Mention-pair decision is difficult

Cluster 1
- Google
- the company

Cluster 2
- Google Plus
- the product

? coreferent

Cluster-pair decision is easier
Clustering-based models

- Current candidate cluster merges depend on previous ones it already made
  - We can’t use supervised learning.
  - Researchers used reinforcement learning to train the model
  - Reward function is the change in a coreference evaluation metric

- Some nice attempts but NOT the dominating approach
Features

• Non-neural coreference models:
  • Person/Number/Gender agreement
    • Jack gave Mary a gift. She was excited.
  • Semantic compatibility
    • ... the mining conglomerate ... the company ...
  • Certain syntactic constraints
    • John bought him a new car. [him can not be John]
  • More recently mentioned entities preferred for referenced
    • John went to a movie. Jack went as well. He was not busy.
  • Grammatical Role: Prefer entities in the subject position
    • John went to a movie with Jack. He was not busy.
  • Parallelism:
    • John went with Jack to a movie. Joe went with him to a bar.
Features

- Neural coreference models:
End-to-end coreference resolution

- A mention-ranking model
- Joint mention detection and clustering — so you don’t need an additional mention detector (parser/part-of-speech tagger)

\[ J = \sum_{i=2}^{N} \left( -\log \left( \sum_{j=1}^{i-1} 1(y_{ij} = 1) p(m_j, m_i) \right) \right) \]

We consider all the possible spans + {NA}

\[ p(m_j, m_i) = \frac{\exp(s(m_j, m_i))}{\sum_{j' < i} \exp(m_{j'}, m_i)} \]

\[ N = \frac{T(T + 1)}{2} + 1 \]

T: number of words

(Lee et al, 2017): End-to-end Neural Coreference Resolution
End-to-end coreference resolution

\[ s(i, j) = s_m(i) + s_m(j) + s_a(i, j) \]

Are spans \(i\) and \(j\) coreferent mentions?
Is \(i\) a mention?
Is \(j\) a mention?
Do they look coreferent?

\[ s_m(i) = \mathbf{w}_m \cdot \text{FFNN}_m(\mathbf{g}_i) \]
\[ s_a(i, j) = \mathbf{w}_a \cdot \text{FFNN}_a([\mathbf{g}_i, \mathbf{g}_j, \mathbf{g}_i \odot \mathbf{g}_j, \phi(i, j)]) \]

Let’s compute a vector representation \(\mathbf{g}_i \in \mathbb{R}^d\) for each span \(i\)

\(\phi(i, j)\): manual features such speaker/gender information

(Lee et al, 2017): End-to-end Neural Coreference Resolution
End-to-end coreference resolution

Span representation ($g$)
Span head ($\hat{x}$)
Bidirectional LSTM ($x^*$)
Word & character embedding ($x$)

Mention score ($s_m$)

Span representation: $g_i = [\mathbf{x}^*_{\text{START}(i)}, \mathbf{x}^*_{\text{END}(i)}, \hat{x}_i, \phi(i)]$

BILSTM hidden states for span’s start and end
Attention-based representation (details next slide) of the words in the span
Additional features

(Lee et al, 2017): End-to-end Neural Coreference Resolution
End-to-end coreference resolution

Attention scores

\[ \alpha_t = \mathbf{w}_\alpha \cdot \text{FFNN}_\alpha(\mathbf{x}_t^*) \]

dot product of weight vector and transformed hidden state

Attention distribution

\[ a_{i,t} = \frac{\exp(\alpha_t)}{\sum_{k=\text{START}(i)}^{\text{END}(i)} \exp(\alpha_k)} \]

just a softmax over attention scores for the span

Final representation

\[ \hat{\mathbf{x}}_i = \sum_{t=\text{START}(i)}^{\text{END}(i)} a_{i,t} \cdot \mathbf{x}_t \]

Attention-weighted sum of word embeddings

\[ g_i = [\mathbf{x}_{\text{START}(i)}^*, \mathbf{x}_{\text{END}(i)}^*, \hat{\mathbf{x}}_i, \phi(i)] \]

hidden states for span’s start and end

Attention-based representation

Represents the context to the left and right of the span

Represents the span itself

Represents other information not in the text

(Lee et al, 2017): End-to-end Neural Coreference Resolution
End-to-end coreference resolution

Testing time:

- $O(T^2)$ spans of text in a document
- $O(T^4)$ possible pairs of spans — too expensive!
- Use $s_m(i)$ for aggressive pruning
Coreference evaluation

- You need to get both “mentions” and “clusters” correctly.
- Standard practice: we use 3 types of metrics
  - B³: mention-based
  - MUC: link-based (pair of mentions)
  - CEAF: entity-based
  - .. and finally take the average of these 3 F1 scores

https://github.com/conll/reference-coreference-scorers
**B-cubed evaluation metric**

- For each mention in the reference chain, compute a precision and a recall (e.g., # of mentions in the same reference chain with the current mention)
- The final precision/recall is an average of all the mentions
System performance

- Evaluation on English Ontonotes (CoNLL-2012 Shared Task)
- \#Train: 2,802 / \#Dev: 343 / \#Test: 348 documents

<table>
<thead>
<tr>
<th></th>
<th>MUC</th>
<th></th>
<th></th>
<th></th>
<th>B³</th>
<th></th>
<th></th>
<th>CEAFΦ₄</th>
<th></th>
<th></th>
<th>Avg. F1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
<td>R</td>
<td>F1</td>
<td>P</td>
<td>R</td>
<td>F1</td>
<td>P</td>
<td>R</td>
<td>F1</td>
<td>Avg. F1</td>
<td></td>
</tr>
<tr>
<td>Lee et al. (2017) (single model)</td>
<td>78.4</td>
<td>73.4</td>
<td>75.8</td>
<td>68.6</td>
<td>61.8</td>
<td>65.0</td>
<td>62.7</td>
<td>59.0</td>
<td>60.8</td>
<td>67.2</td>
<td></td>
</tr>
<tr>
<td>Clark and Manning (2016a)</td>
<td>79.2</td>
<td>70.4</td>
<td>74.6</td>
<td>69.9</td>
<td>58.0</td>
<td>63.4</td>
<td>63.5</td>
<td>55.5</td>
<td>59.2</td>
<td>65.7</td>
<td></td>
</tr>
<tr>
<td>Clark and Manning (2016b)</td>
<td>79.9</td>
<td>69.3</td>
<td>74.2</td>
<td>71.0</td>
<td>56.5</td>
<td>63.0</td>
<td>63.8</td>
<td>54.3</td>
<td>58.7</td>
<td>65.3</td>
<td></td>
</tr>
<tr>
<td>Wiseman et al. (2016)</td>
<td>77.5</td>
<td>69.8</td>
<td>73.4</td>
<td>66.8</td>
<td>57.0</td>
<td>61.5</td>
<td>62.1</td>
<td>53.9</td>
<td>57.7</td>
<td>64.2</td>
<td></td>
</tr>
<tr>
<td>Wiseman et al. (2015)</td>
<td>76.2</td>
<td>69.3</td>
<td>72.6</td>
<td>66.2</td>
<td>55.8</td>
<td>60.5</td>
<td>59.4</td>
<td>54.9</td>
<td>57.1</td>
<td>63.4</td>
<td></td>
</tr>
<tr>
<td>Fernandes et al. (2014)</td>
<td>75.9</td>
<td>65.8</td>
<td>70.5</td>
<td>77.7</td>
<td>65.8</td>
<td>71.2</td>
<td>43.2</td>
<td>55.0</td>
<td>48.4</td>
<td>63.4</td>
<td></td>
</tr>
<tr>
<td>Clark and Manning (2015)</td>
<td>76.1</td>
<td>69.4</td>
<td>72.6</td>
<td>65.6</td>
<td>56.0</td>
<td>60.4</td>
<td>59.4</td>
<td>53.0</td>
<td>56.0</td>
<td>63.0</td>
<td></td>
</tr>
<tr>
<td>Martschat and Strube (2015)</td>
<td>76.7</td>
<td>68.1</td>
<td>72.2</td>
<td>66.1</td>
<td>54.2</td>
<td>59.6</td>
<td>59.5</td>
<td>52.3</td>
<td>55.7</td>
<td>62.5</td>
<td></td>
</tr>
<tr>
<td>Durrett and Klein (2014)</td>
<td>72.6</td>
<td>69.9</td>
<td>71.2</td>
<td>61.2</td>
<td>56.4</td>
<td>58.7</td>
<td>56.2</td>
<td>54.2</td>
<td>55.2</td>
<td>61.7</td>
<td></td>
</tr>
<tr>
<td>Björkelund and Kuhn (2014)</td>
<td>74.3</td>
<td>67.5</td>
<td>70.7</td>
<td>62.7</td>
<td>55.0</td>
<td>58.6</td>
<td>59.4</td>
<td>52.3</td>
<td>55.6</td>
<td>61.6</td>
<td></td>
</tr>
<tr>
<td>Durrett and Klein (2013)</td>
<td>72.9</td>
<td>65.9</td>
<td>69.2</td>
<td>63.6</td>
<td>52.5</td>
<td>57.5</td>
<td>54.3</td>
<td>54.4</td>
<td>54.3</td>
<td>60.3</td>
<td></td>
</tr>
</tbody>
</table>

“Eash Victories and Uphill Battles in Coreference Resolution”
System performance

Avg. F1

67.2 (Lee et al, 2017)
71.0 (Lee et al, 2018)
73.0 (Joshi & Chen, 2019)
77.1 (Lee et al, 2018)
79.6 (SpanBERT)