Coreference Resolution

Daniel Biro, Joel Lee, Louis, Ding Feng, Mohit

1. Introduction

Daniel Biro

What is 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.

What is 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.

What is 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.

What is Coreference Resolution?

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

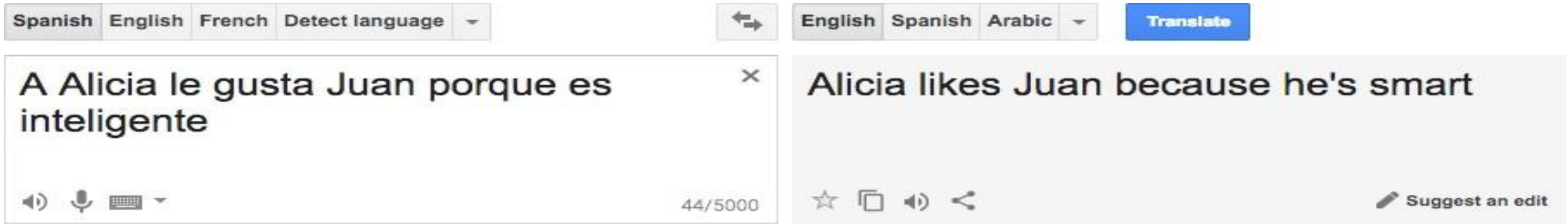
Barack Obama nominated Hillary Rodham Clinton as his secretary of state on Monday. He chose her because she had foreign affairs experience as a former First Lady.

Applications

- Full text understanding
  - information extraction, question answering, summarization, …
  - “He was born in 1961”

Applications

- Full text understanding
- Machine translation
  - languages have different features for gender, number, dropped pronouns, etc.



Applications

- Full text understanding
- Machine translation
- Dialogue Systems

  "Book tickets to see James Bond"

  "Spectre is playing near you at 2:00 and 3:00 today. How many tickets would you like?"

  "Two tickets for the showing at three"

Coreference Resolution is Really Difficult!

- "She poured water from the pitcher into the cup until it was full"
- "She poured water from the pitcher into the cup until it was empty"

- The trophy would not fit in the suitcase because it was too big.
- The trophy would not fit in the suitcase because it was too small.

- These are called **Winograd Schema**
  - Recently proposed as an alternative to the Turing test
  - If you've fully solved coreference, arguably you've solved AI

Coreference Resolution in Two Steps

1. Detect the mentions (easy)

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

   - mentions can be nested!

2. Cluster the mentions (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, etc.
2. Named entities
    - People, places, etc.
3. Noun phrases
    - "a dog," "the big fluffy cat stuck in the tree"

Mention Detection

- Span of text referring to some entity
- For detection: use other NLP systems

1. Pronouns
    - Use a part-of-speech tagger
2. Named entities
    - Use a NER system
3. Noun phrases
    - Use a constituency parser

Mention Detection: Not so Simple

- Marking all pronouns, named entities, and NPs as mentions over-generates mentions
- Are these mentions?
  - It is sunny
  - Every student
  - No student
  - The best donut in the world
  - 100 miles
- Some gray area in defining “mention”: have to pick a convention and go with it

How to deal with these bad mentions?

- Could train a classifier to filter out spurious mentions
- Much more common: keep all mentions as “candidate mentions”
  - After your coreference system is done running discard all singleton mentions (i.e., ones that have not been marked as coreference with anything else)

Can we avoid a pipelined system?

- We could instead train a classifier specifically for mention detection instead of using a POS tagger, NER system, and parser.

- Or even jointly do mention-detection and coreference resolution end-to-end instead of in two steps

First, some linguistics

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

- Another kind of reference is **anaphora:** when a term (anaphor) refers to another term (antecedent) and the interpretation of the anaphor is in some way determined by the interpretation of the antecedent
  - Barack Obama said he would sign the bill.

    antecedent anaphor

Anaphora vs Coreference

- Coreference with named entities

text Barack Obama Obama

world

- Anaphora

text Barack Obama he

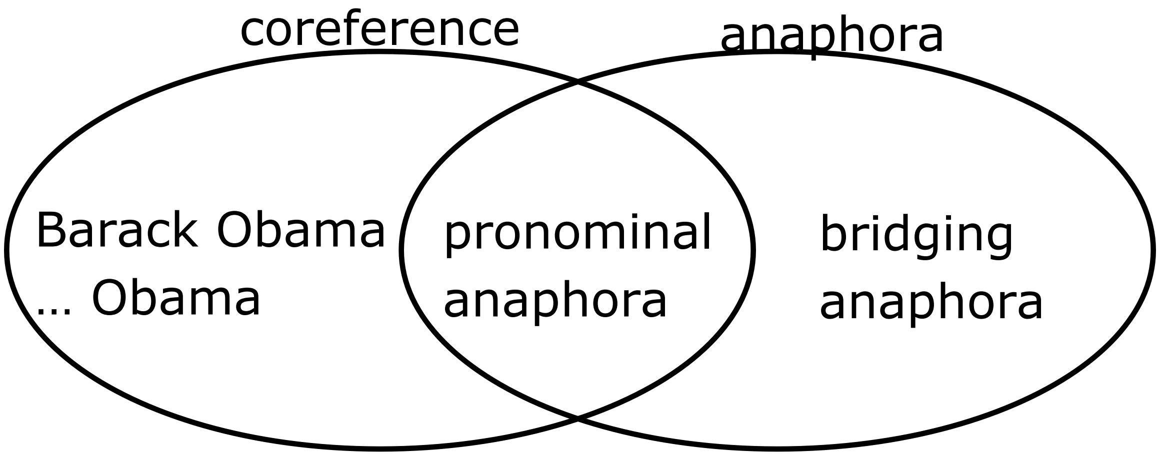
world

Anaphora vs. Coreference

- Not all anaphoric relations are coreferential

We went to see a concert last night. The tickets were really expensive.

- This is referred to as bridging anaphora.



Cataphora

- Usually the antecedent comes before the anaphor (e.g., a pronoun), but not always

Cataphora

"From the corner of the divan of Persian saddle- bags on which he was lying, smoking, as was his custom, innumerable cigarettes, Lord Henry Wotton could just catch the gleam of the honey- sweet and honey-coloured blossoms of a laburnum…"

(Oscar Wilde – The Picture of Dorian Gray)

2. Coreference Models

Louis - Joel - Ding Feng

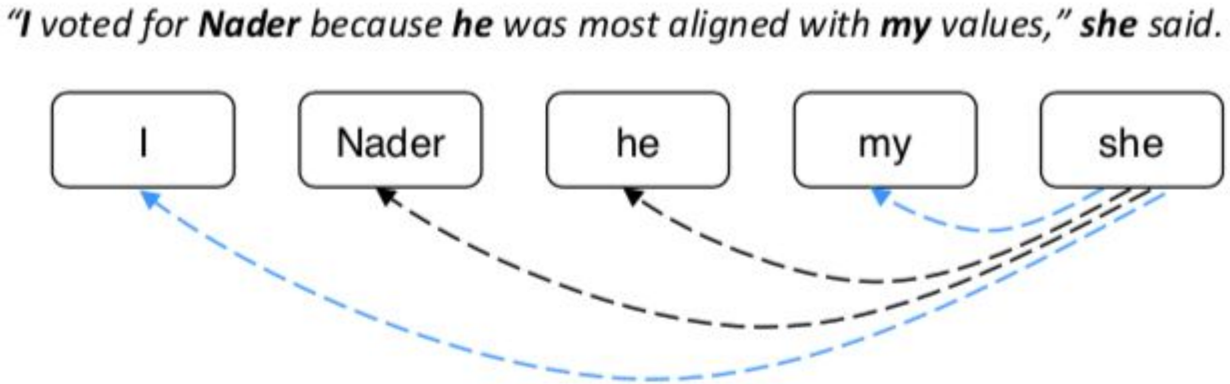
Kinds of Coreference Models

2.1 Mention Pair Model

2.2 Mention Ranking Model

2.3 Clustering Model

2.1 Mention Pair - Method



Method:

1. Mention Detection
2. Coreferent calculation $p(m_i, m_j)$ for every pair
3. Add coreferent link if $p(m_i, m_j) >$ threshold

2.1 Mention Pair - Training

- N mentions in a document
- $y_{ij} = 1$ if mentions m_i and m_j are coreferent, -1 if otherwise
- Just train with regular cross-entropy loss (looks a bit different because it is binary classification)

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

Iterate through mentions

Iterate through candidate antecedents (previously occurring mentions)

Coreferent mentions pairs should get high probability, others should get low probability

CROSS-ENTROPY

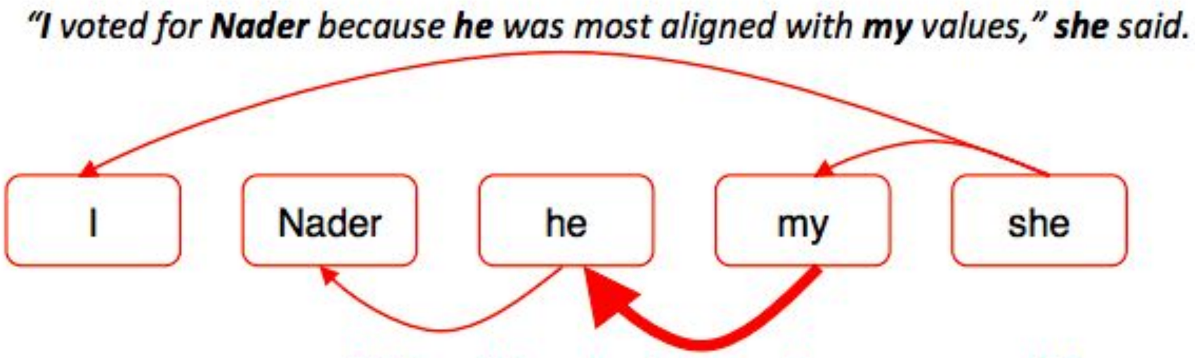
$S(Y)$ L

$$\begin{bmatrix} 0.7 \\ 0.2 \\ 0.1 \end{bmatrix} \quad D(S, L) = -\sum_i L_i \log(S_i) \quad \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \end{bmatrix}$$

$D(S,L) \neq D(L,S)$

2.1 Mention Pair - Limitation

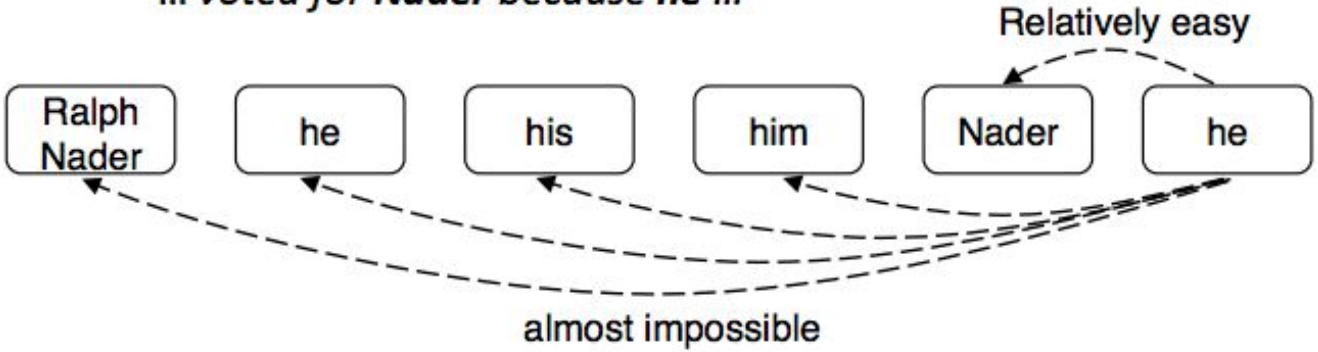
- 1 wrong coreferent link would merge everything



2.1 Mention Pair - Limitation

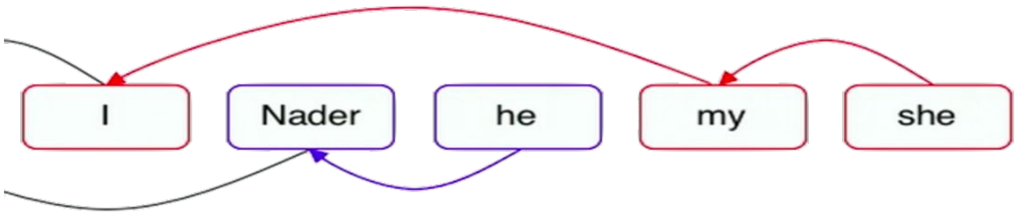
- Inability to process long document

- Suppose we have a long document with the following mentions
  - **Ralph Nader** … **he** … **his** … **him** … <several paragraphs> … *voted for* ***Nader*** *because* ***he*** *…*

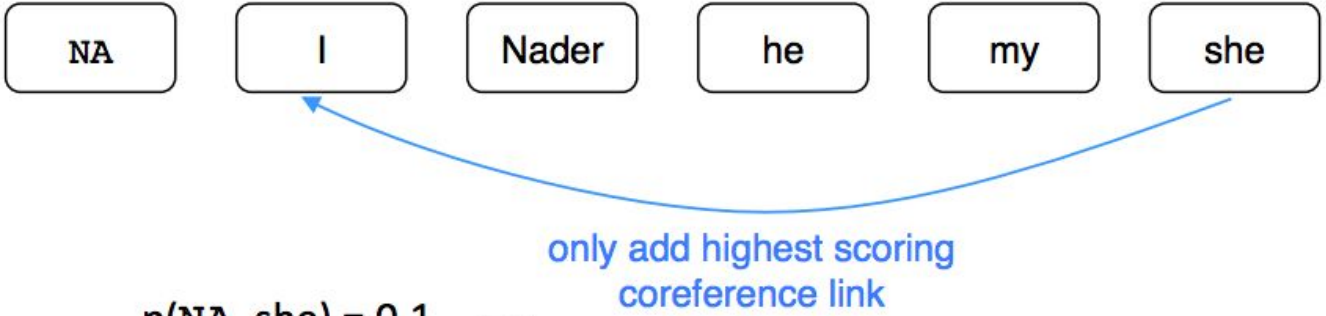


2.2 Mention Ranking - Method

- Only keep highest score coreferent link
- Infer global structure by making a sequence of local decisions



2.2 Mention Ranking - Method



p(NA, she) = 0.1
p(I, she) = 0.5
p(Nader, she) = 0.1
p(he, she) = 0.1
p(my, she) = 0.2

Apply a softmax over the scores for candidate antecedents so probabilities sum to 1

2.2 Mention Ranking - Training

- Coreferent Likelihood Score

$$\sum_{j=1}^{i-1} \mathbb{1}(y_{ij} = 1)\,p(m_j, m_i)$$

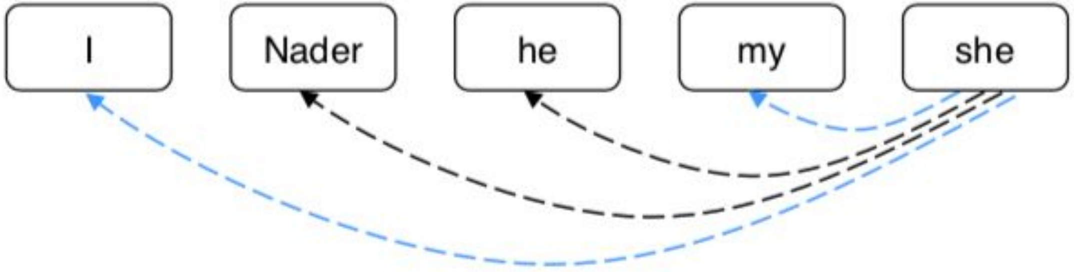
Iterate through candidate antecedents (previously occurring mentions)

For ones that are coreferent to m_j...

...we want the model to assign a high probability

p(NA, she) = 0.1
p(I, she) = 0.5
p(Nader, she) = 0.1
p(he, she) = 0.1
p(my, she) = 0.2

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



2.2 Mention Ranking - Training

- Mathematically, we want to maximize this probability:

$$\sum_{j=1}^{i-1} \mathbb{1}(y_{ij} = 1)p(m_j, m_i)$$

- Turning this into a loss function:

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

Iterate over all the mentions in the document

Usual trick of taking negative log to go from likelihood to loss

Coreferent Score - $p(m_i, m_j)$

Statistical classifier & Simple Neural Network

- Joel Lee -

1. Non-Neural Coref Model: Features

- 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.
- …

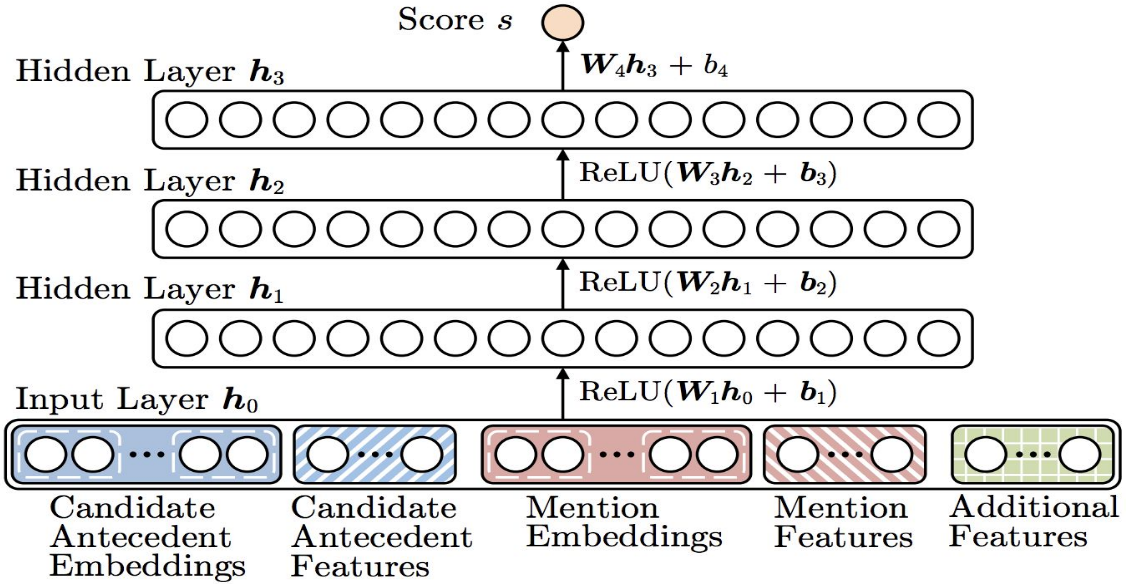
(1) Separately, Clinton transition officials said that *Frank Newman*, 50, *vice chairman* and chief financial officer of BankAmerica Corp., is expected to be nominated as assistant Treasury secretary for domestic finance.

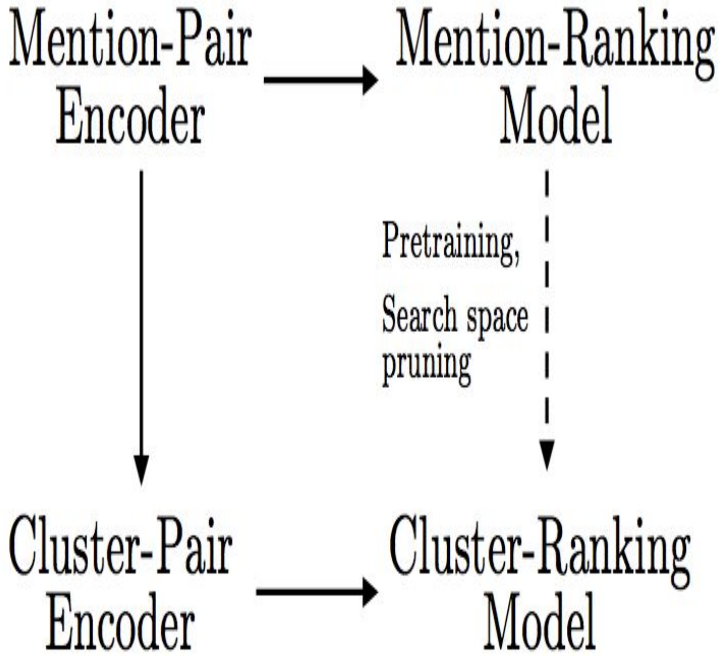
Table 1
Feature vector of the markable pair (i = *Frank Newman*, j = *vice chairman*).

Feature	Value	Comments
DIST	0	i and j are in the same sentence
I_PRONOUN	–	i is not a pronoun
J_PRONOUN	–	j is not a pronoun
STR_MATCH	–	i and j do not match
DEF_NP	–	j is not a definite noun phrase
DEM_NP	–	j is not a demonstrative noun phrase
NUMBER	+	i and j are both singular
SEMCLASS	1	i and j are both persons (This feature has three values: false(0), true(1), unknown(2).)
GENDER	1	i and j are both males (This feature has three values: false(0), true(1), unknown(2).)
PROPER_NAME	–	Only i is a proper name
ALIAS	–	j is not an alias of i
APPOSITIVE	+	j is in apposition to i

2. Neural Coref Model

- Standard feed-forward neural network
  - Input layer: word embeddings and a few categorical features





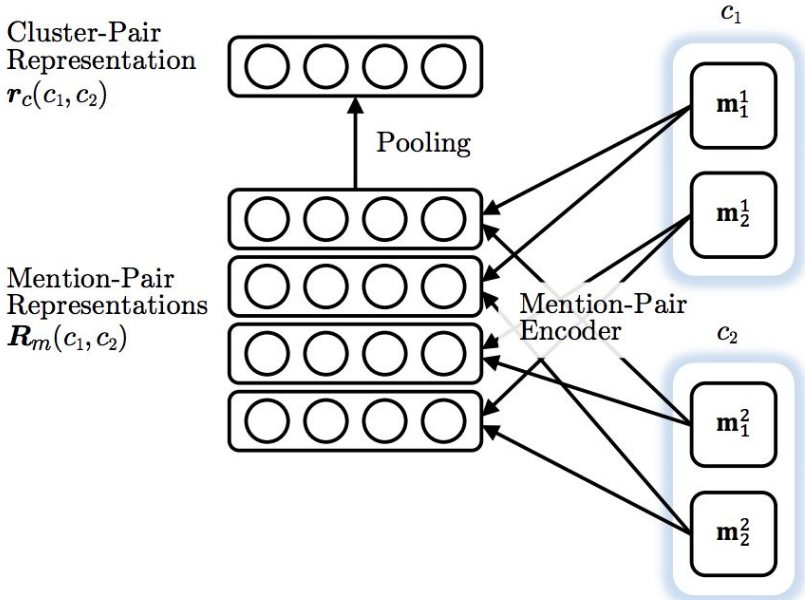


Figure 3: Cluster-pair encoder.

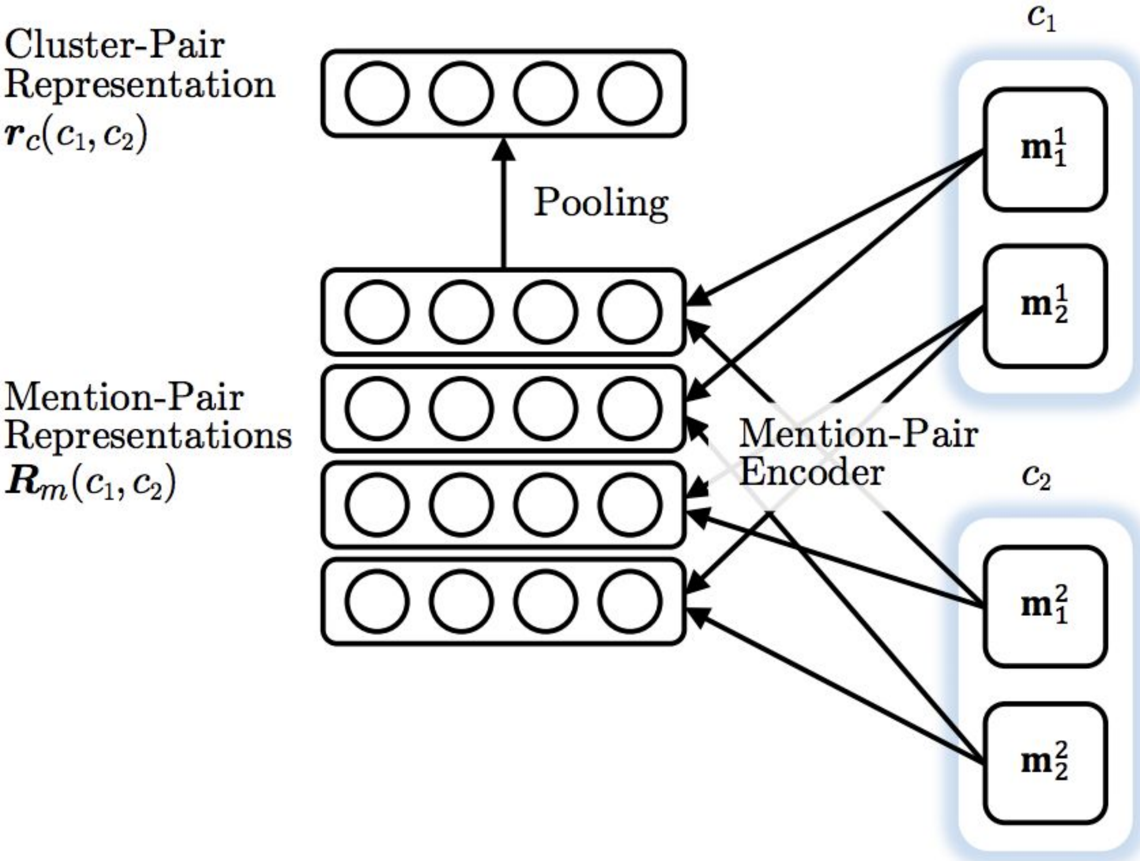


Figure 3: Cluster-pair encoder.

2. Neural Coref Model: Inputs

- Embeddings
  - Previous two words, first word, last word, head word, … of each mention
    - The **head** word is the "most important" word in the mention – you can find it using a parser. e.g., *The fluffy **cat** stuck in the tree*
- Still need some other features:
  - Distance
  - Document genre
  - Speaker information

2.3 End-to-End Coreference Resolution

Ding Feng

End-to-End Neural Coreference Resolution

- Based on Current state-of-the-art model for coreference resolution (Lee et al. EMNLP 2017) https://github.com/kentonl/e2e-coref
- Why interesting?
    - Previous methods offer great performance, built on top of parse trees
        - Hand engineered features
            - Parsing mistakes cascading errors
            - Not generalisable

Previous methods, not end-to-end

Input document -> parser -> engineering -> mentions -> coref

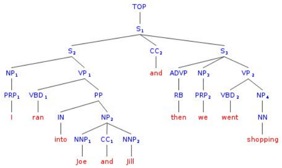
Lorem ipsum dolor

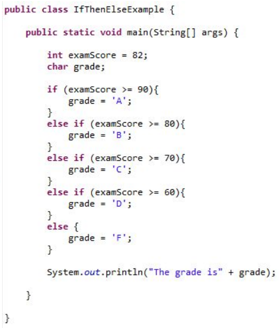
sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt

eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure

aliquip ex ea commodo consequat.

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Lorem ipsum dolor sit amet, consectetur adipiscing

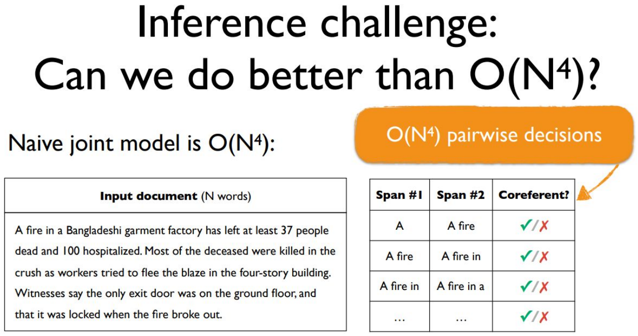




```java
public class IfThenElseExample {

    public static void main(String[] args) {

        int examScore = 82;
        char grade;

        if (examScore >= 90){
            grade = 'A';
        }
        else if (examScore >= 80){
            grade = 'B';
        }
        else if (examScore >= 70){
            grade = 'C';
        }
        else if (examScore >= 60){
            grade = 'D';
        }
        else {
            grade = 'F';
        }

        System.out.println("The grade is" + grade);

    }

}
```

End-to-end approach

- Joint mention detection and clustering
- No preprocessing, parsing etc.
- How?
    - Consider All possible spans up to size L=10, calculate a coreference score, S(i,j)
        - Learn Rank antecedent spans
        - Factored model to prune



Span Ranking

- Reason over all possible spans
- Assign an antecedent to every span

	Span	Antecedent
1	A	y_1
2	A fire	y_2
3	A fire in	y_3
…	…	…
M	out	y_M

$$y_3 \in \{\epsilon, 1, 2\}$$

Coreference link from span 2 to span 3

Example Clustering

Input document
A fire in a Bangladeshi garment factory has left at least 37 people dead and 100 hospitalized. Most of the deceased were killed in the crush as workers tried to flee the blaze in the four-story building. Witnesses say the only exit door was on the ground floor, and that it was locked when the fire broke out.

Span	Antecedent (y_i)
A	ϵ
A fire	ϵ
...	...
a Bangladeshi garment factory	ϵ
...	...
the four-story building	a Bangladeshi garment factory
...	...
out	ϵ

Learning

Marginal log-likelihood objective.

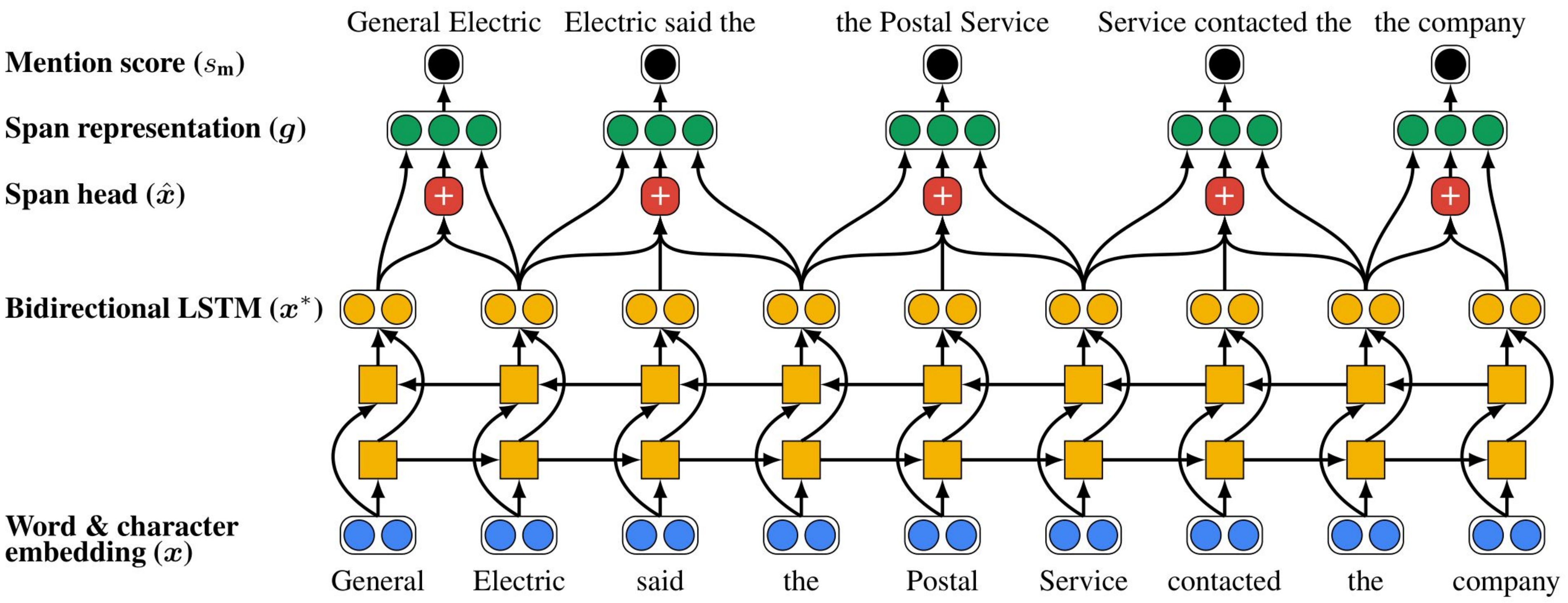
$$\log \prod_{i=1}^{M} \sum_{\hat{y} \in \mathcal{Y}(i) \cap \textsc{gold}(i)} P(\hat{y} \mid D)$$

- Related to Durrett & Klein (2013)
- Model can assign credit/blame to the mention or antecedent factors

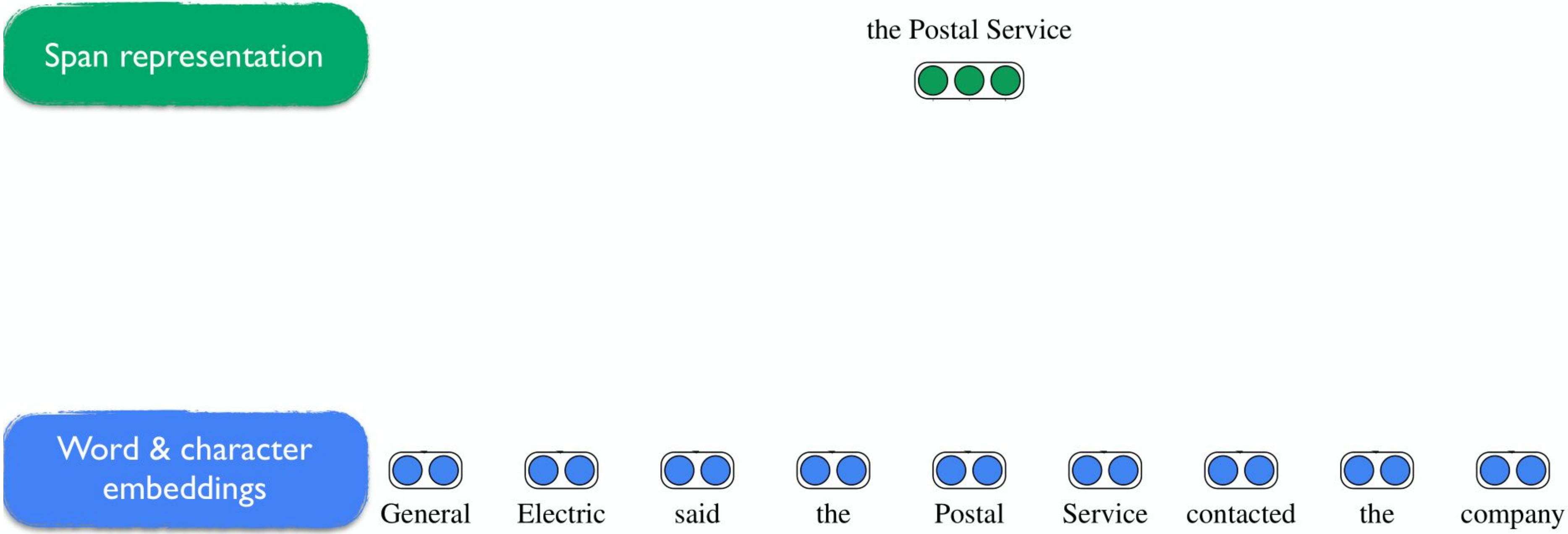
$$s(i,j) = \begin{cases} s_{\mathrm{m}}(i) + s_{\mathrm{m}}(j) + s_{\mathrm{a}}(i,j) & j \neq \epsilon \\ 0 & j = \epsilon \end{cases}$$

A fire in a Bangladeshi garment factory has left at least 37 people dead and 100 hospitalized.

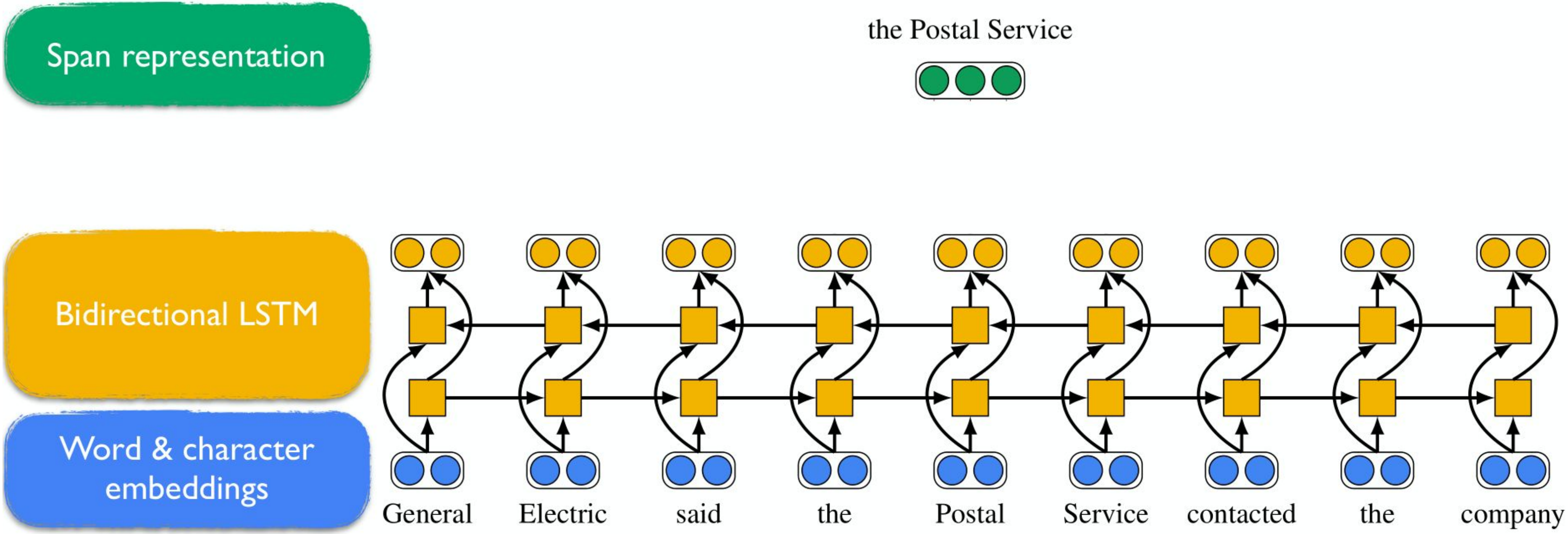
Partial label, only long spans are labeled, how to deal with short ones?


General Electric
Electric said the
the Postal Service
Service contacted the
the company
Mention score (s_m)
Span representation (g)
Span head ($\hat{x}$)
Bidirectional LSTM (x^*)
Word & character embedding (x)
General
Electric
said
the
Postal
Service
contacted
the
company

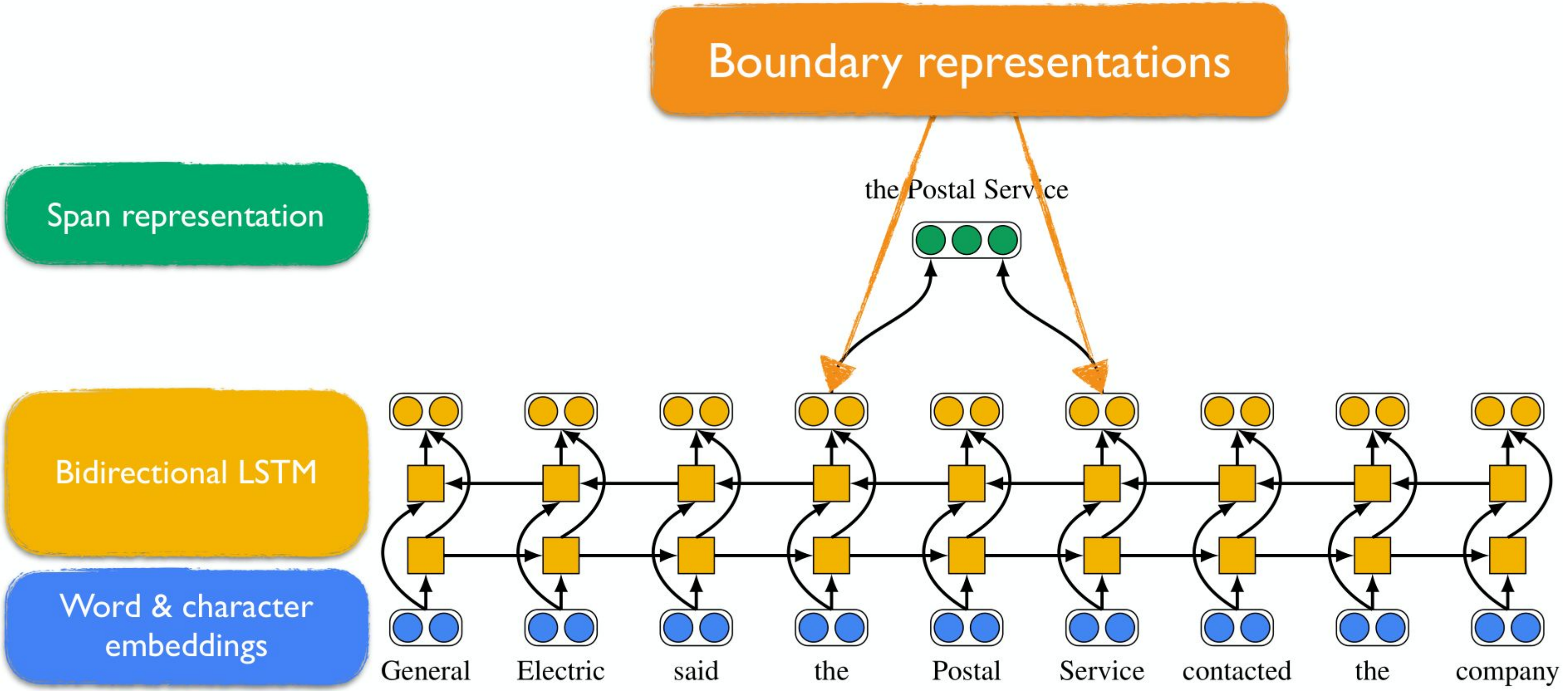
Neural Span Representations



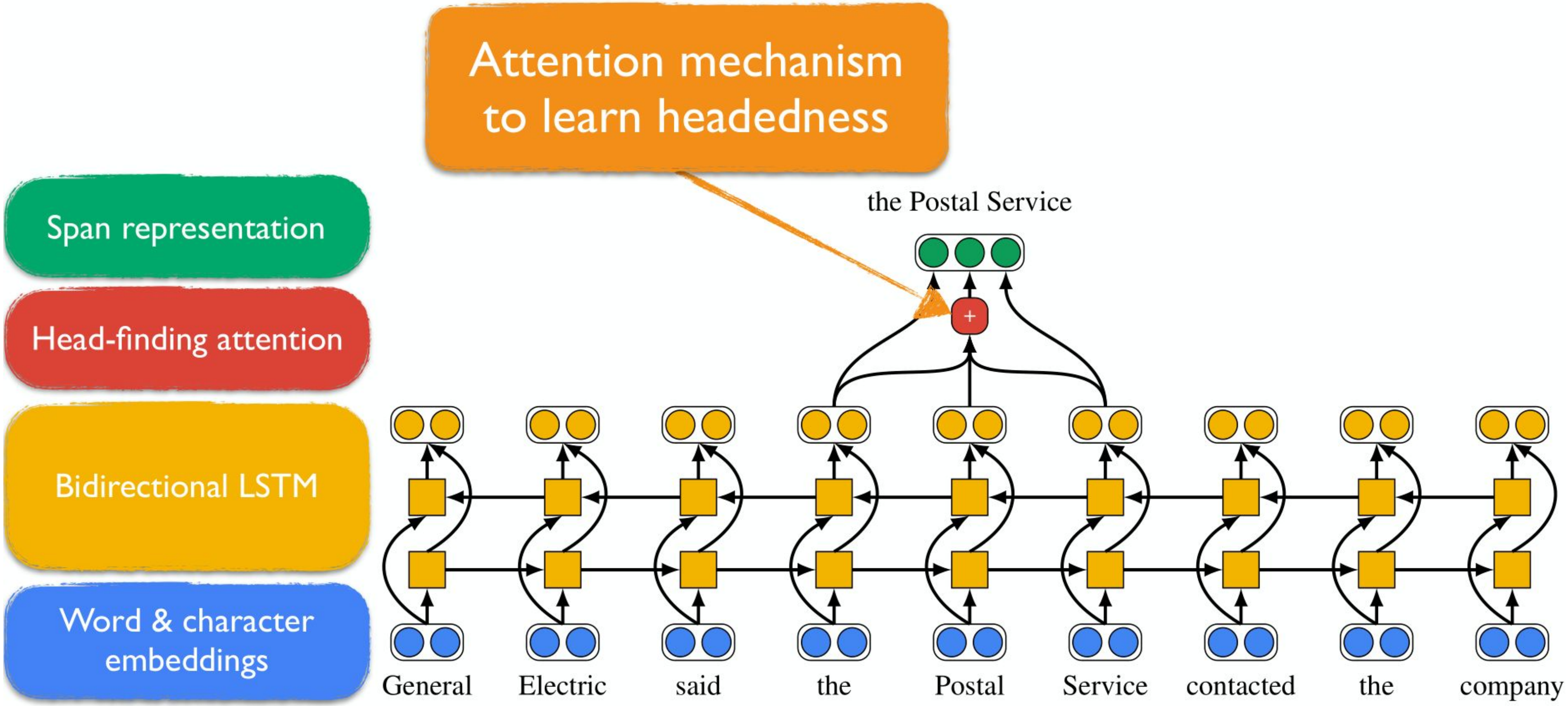
Neural Span Representations



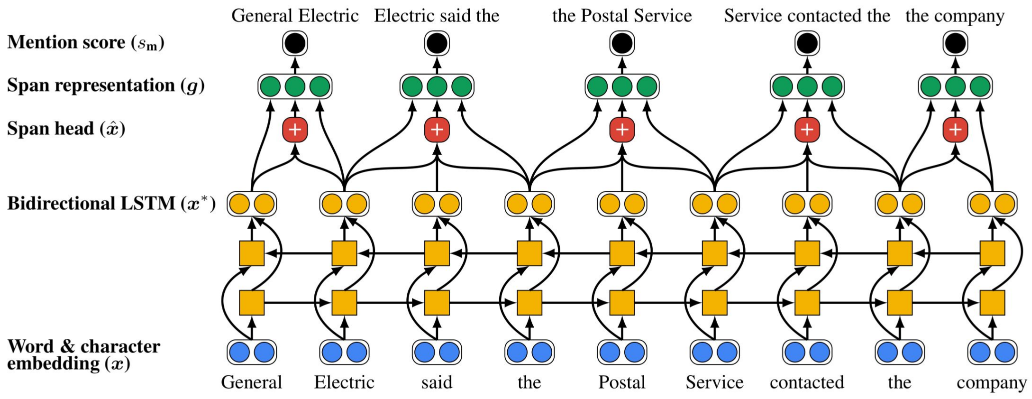
Neural Span Representations



Neural Span Representations



Attention to learn headedness

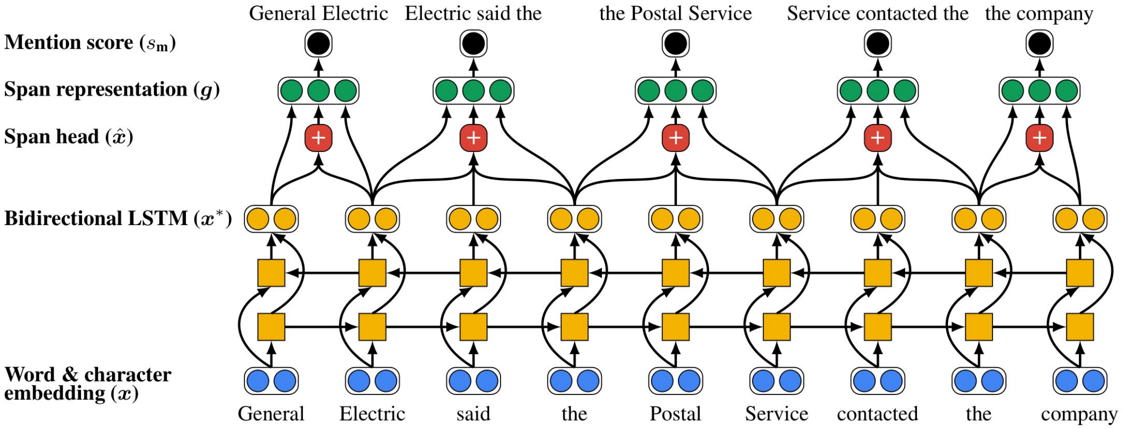


$$\alpha_t = \boldsymbol{w}_\alpha \cdot \mathrm{FFNN}_\alpha(\boldsymbol{x}_t^*)$$

$$a_{i,t} = \frac{\exp(\alpha_t)}{\sum_{k=\mathrm{START}(i)}^{\mathrm{END}(i)} \exp(\alpha_k)}$$

$$\hat{\boldsymbol{x}}_i = \sum_{t=\mathrm{START}(i)}^{\mathrm{END}(i)} a_{i,t} \cdot \boldsymbol{x}_t$$

Attention to learn headedness



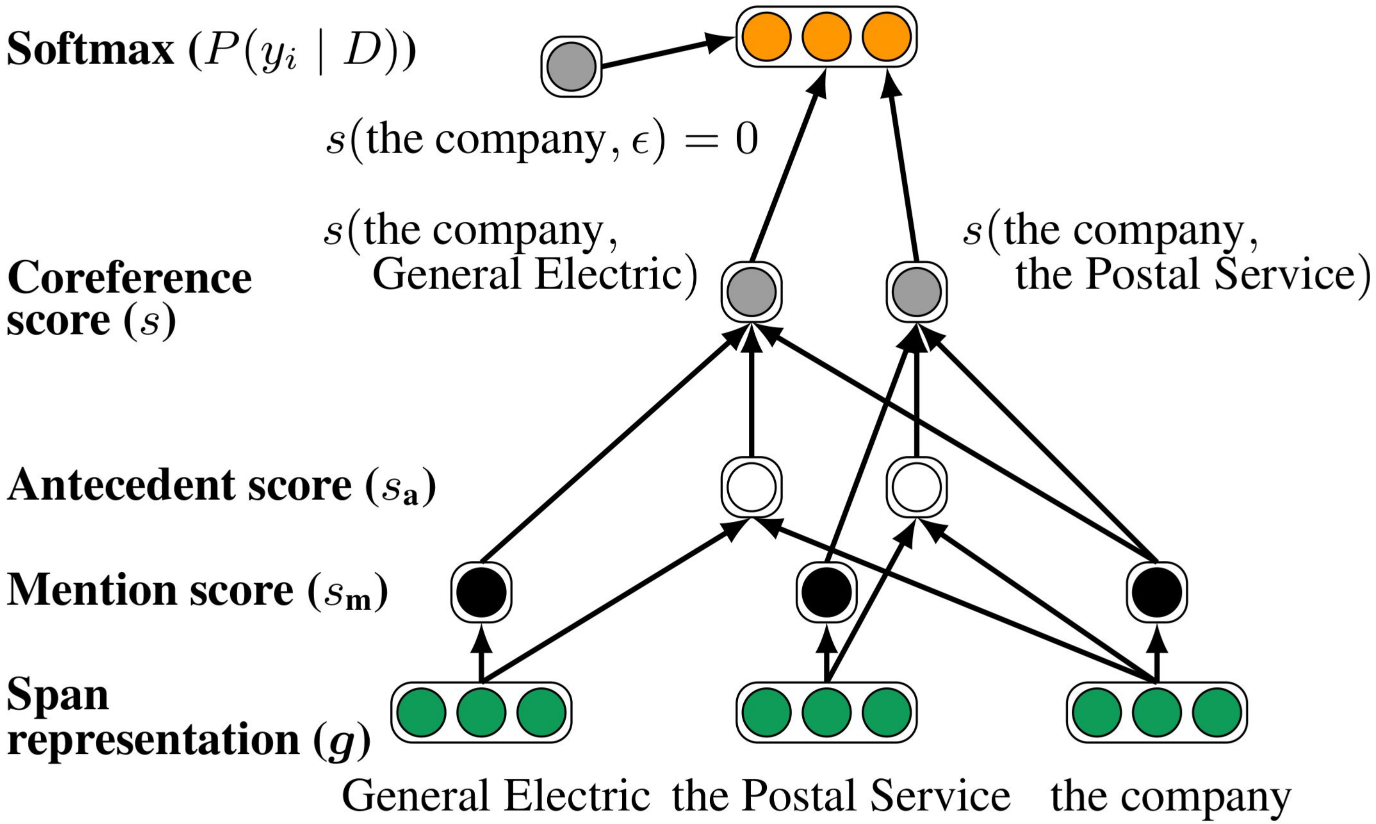
$$\alpha_t = \boldsymbol{w}_\alpha \cdot \mathrm{FFNN}_\alpha(\boldsymbol{x}_t^*)$$

$$a_{i,t} = \frac{\exp(\alpha_t)}{\sum_{k=\mathrm{START}(i)}^{\mathrm{END}(i)} \exp(\alpha_k)}$$

$$\hat{\boldsymbol{x}}_i = \sum_{t=\mathrm{START}(i)}^{\mathrm{END}(i)} a_{i,t} \cdot \boldsymbol{x}_t$$

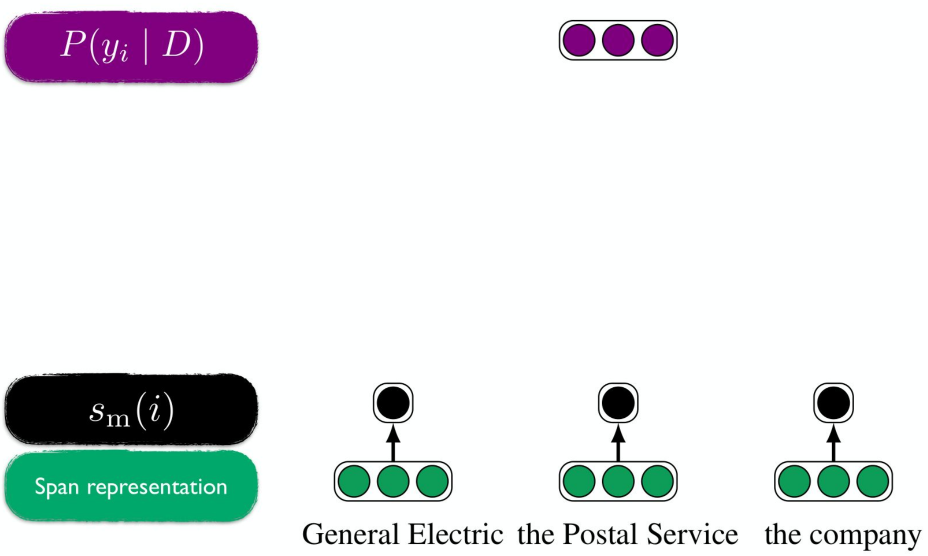
$$\boldsymbol{g}_i = [\boldsymbol{x}^*_{\mathrm{START}(i)}, \boldsymbol{x}^*_{\mathrm{END}(i)}, \hat{\boldsymbol{x}}_i, \phi(i)]$$

Coreference Architecture



Compute Single mention scores

Coreference Architecture



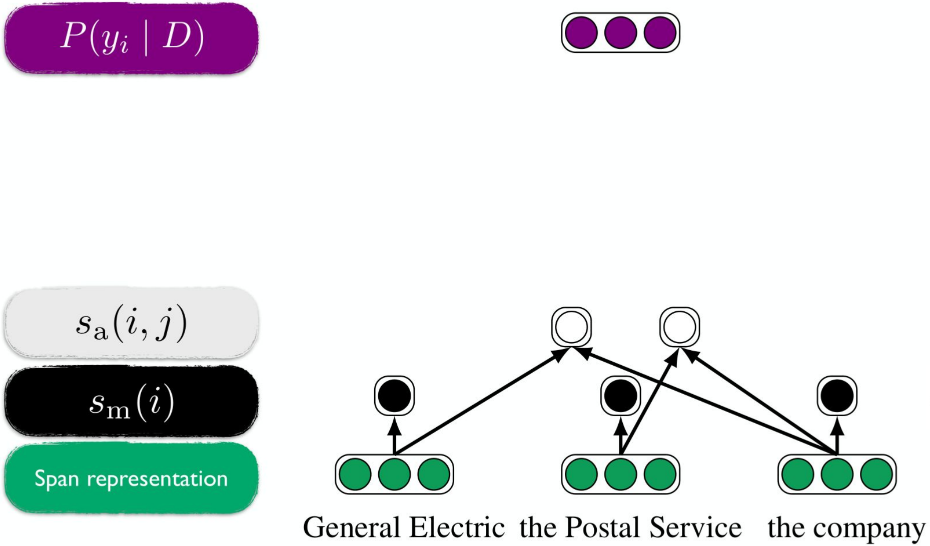
$$s(i,j) = \begin{cases} 0 & j = \epsilon \\ s_\mathrm{m}(i) + s_\mathrm{m}(j) + s_\mathrm{a}(i,j) & j \neq \epsilon \end{cases}$$

$$\boldsymbol{g}_i = [\boldsymbol{x}^*_{\mathrm{START}(i)}, \boldsymbol{x}^*_{\mathrm{END}(i)}, \hat{\boldsymbol{x}}_i, \phi(i)]$$

$$s_\mathrm{m}(i) = \boldsymbol{w}_\mathrm{m} \cdot \mathrm{FFNN}_\mathrm{m}(\boldsymbol{g}_i)$$

Compute Antecedent mention scores

Coreference Architecture



$$s(i,j) = \begin{cases} 0 & j = \epsilon \\ s_m(i) + s_m(j) + s_a(i,j) & j \neq \epsilon \end{cases}$$

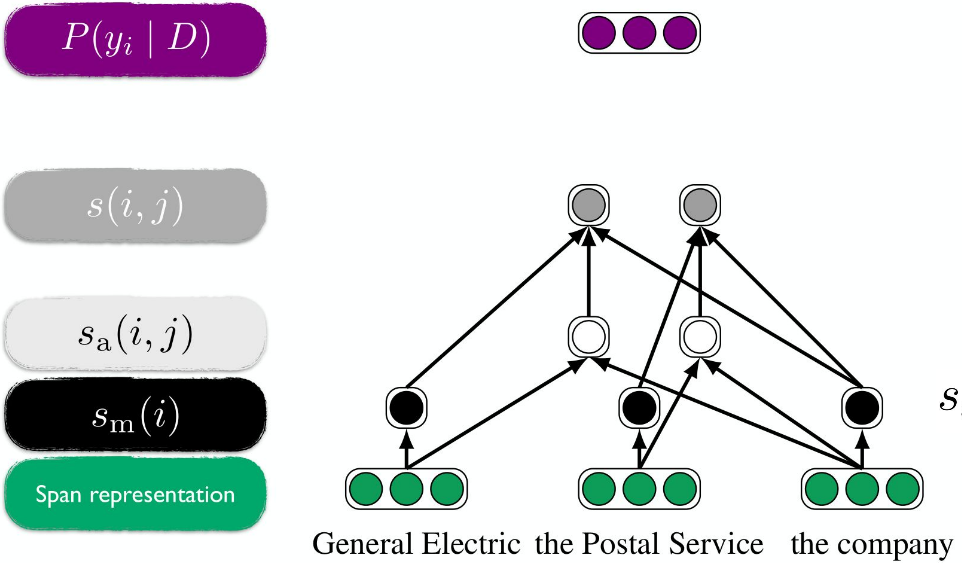
$$\boldsymbol{g}_i = [\boldsymbol{x}^*_{\mathrm{START}(i)}, \boldsymbol{x}^*_{\mathrm{END}(i)}, \hat{\boldsymbol{x}}_i, \phi(i)]$$

$$s_{\mathrm{m}}(i) = \boldsymbol{w}_{\mathrm{m}} \cdot \mathrm{FFNN}_{\mathrm{m}}(\boldsymbol{g}_i)$$

$$s_{\mathrm{a}}(i,j) = \boldsymbol{w}_{\mathrm{a}} \cdot \mathrm{FFNN}_{\mathrm{a}}([\boldsymbol{g}_i, \boldsymbol{g}_j, \boldsymbol{g}_i \circ \boldsymbol{g}_j, \phi(i,j)])$$

Combine the scores

Coreference Architecture



$$s(i,j) = \begin{cases} 0 & j = \epsilon \\ s_\mathrm{m}(i) + s_\mathrm{m}(j) + s_\mathrm{a}(i,j) & j \neq \epsilon \end{cases}$$

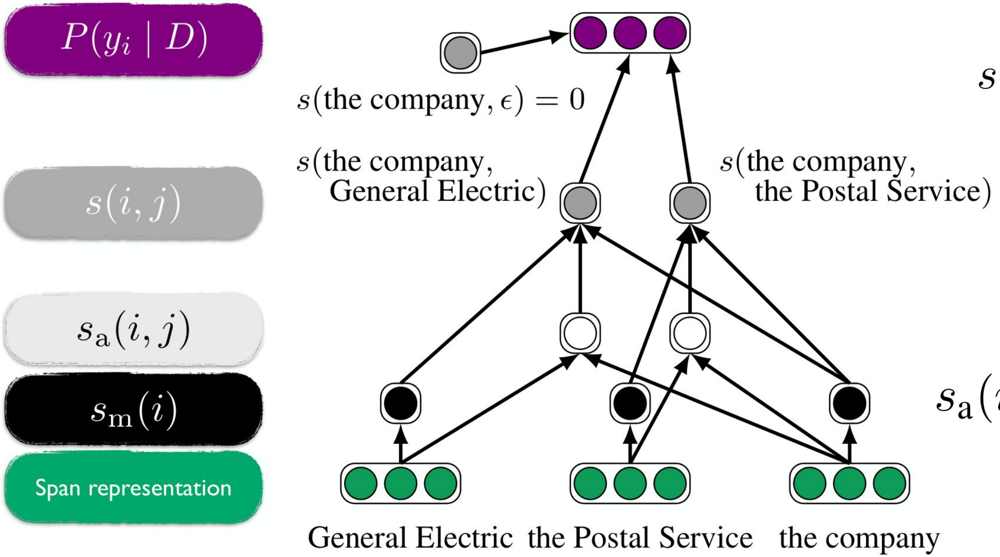
$$\boldsymbol{g}_i = [\boldsymbol{x}^*_{\mathrm{START}(i)}, \boldsymbol{x}^*_{\mathrm{END}(i)}, \hat{\boldsymbol{x}}_i, \phi(i)]$$

$$s_\mathrm{m}(i) = \boldsymbol{w}_\mathrm{m} \cdot \mathrm{FFNN}_\mathrm{m}(\boldsymbol{g}_i)$$

$$s_\mathrm{a}(i,j) = \boldsymbol{w}_\mathrm{a} \cdot \mathrm{FFNN}_\mathrm{a}([\boldsymbol{g}_i, \boldsymbol{g}_j, \boldsymbol{g}_i \circ \boldsymbol{g}_j, \phi(i,j)])$$

Softmax

Coreference Architecture



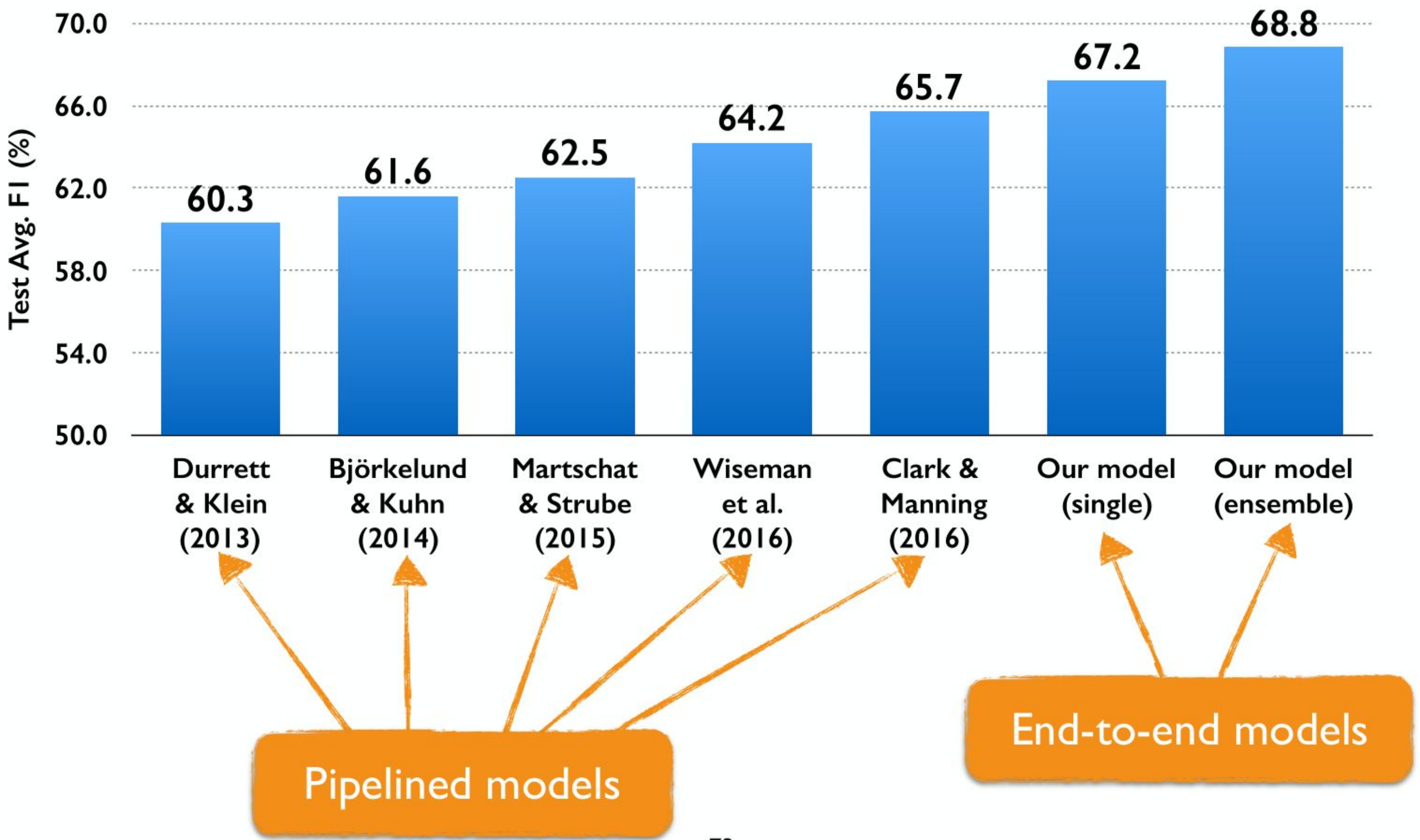
$$s(i, j) = \begin{cases} 0 & j = \epsilon \\ s_m(i) + s_m(j) + s_a(i, j) & j \neq \epsilon \end{cases}$$

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

$$s_m(i) = \boldsymbol{w}_m \cdot \text{FFNN}_m(\boldsymbol{g}_i)$$

$$s_a(i, j) = \boldsymbol{w}_a \cdot \text{FFNN}_a([\boldsymbol{g}_i, \boldsymbol{g}_j, \boldsymbol{g}_i \circ \boldsymbol{g}_j, \phi(i, j)])$$

Coreference Results



Qualitative Analysis

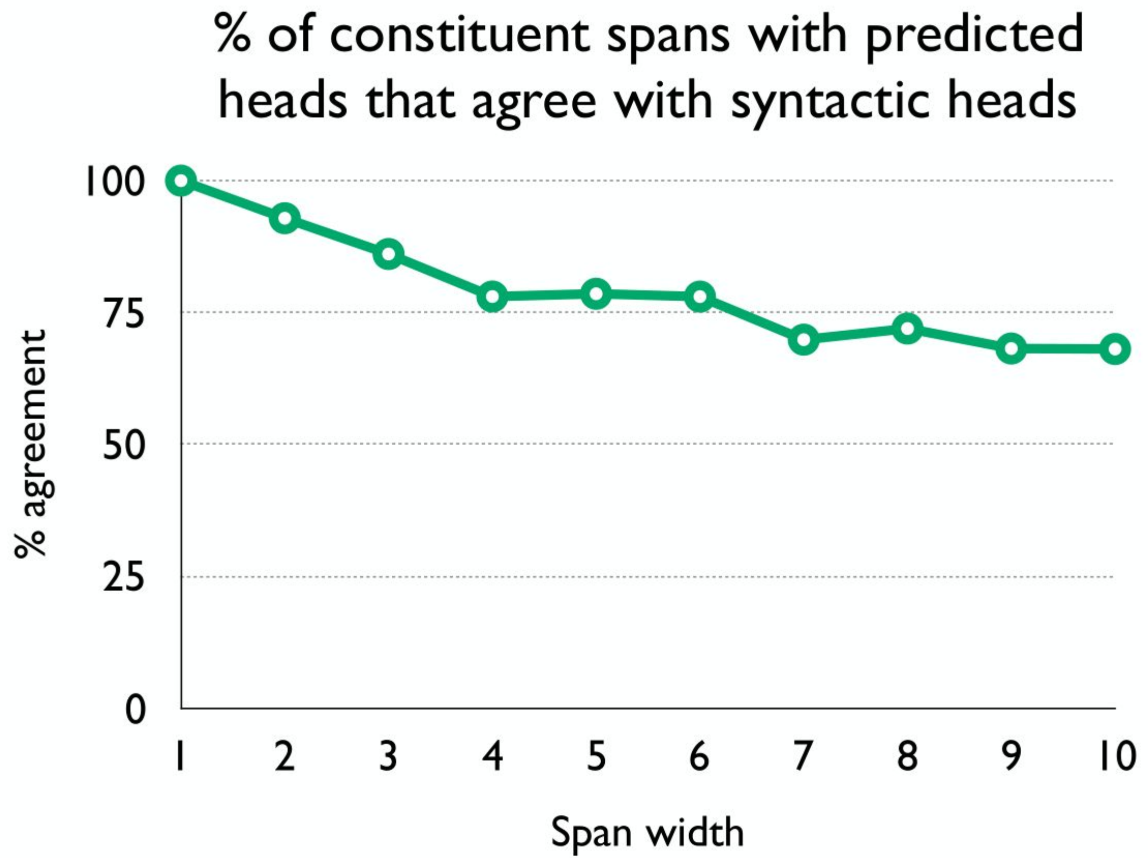
: Mention in

: Head-findin

Attention-based head finder facilitates soft similarity cues

A fire in a Bangladeshi garment factory has left at least 37 people dead and 100 hospitalized. Most of the deceased were killed in the crush as workers tried to flee the blaze in the four-story building.

Head-finding Agreement



Common Error Case

: Mention in a predicted cluster

: Head-finding attention weight

The flight attendants have until 6:00 today to ratify labor concessions. The pilots' union and ground crew did so yesterday.

Conflating **relatedness** with **paraphrasing**

Conclusion

- State-of-the-art end-to-end coreference resolver
  - Scalable inference
  - Learns latent mentions and heads
  - https://github.com/kentonl/e2e-coref
- Relatively simplistic model:
  - Doesn’t explicitly model clusters
  - Lacks discourse reasoning and world knowledge
  - Still a long way to go!

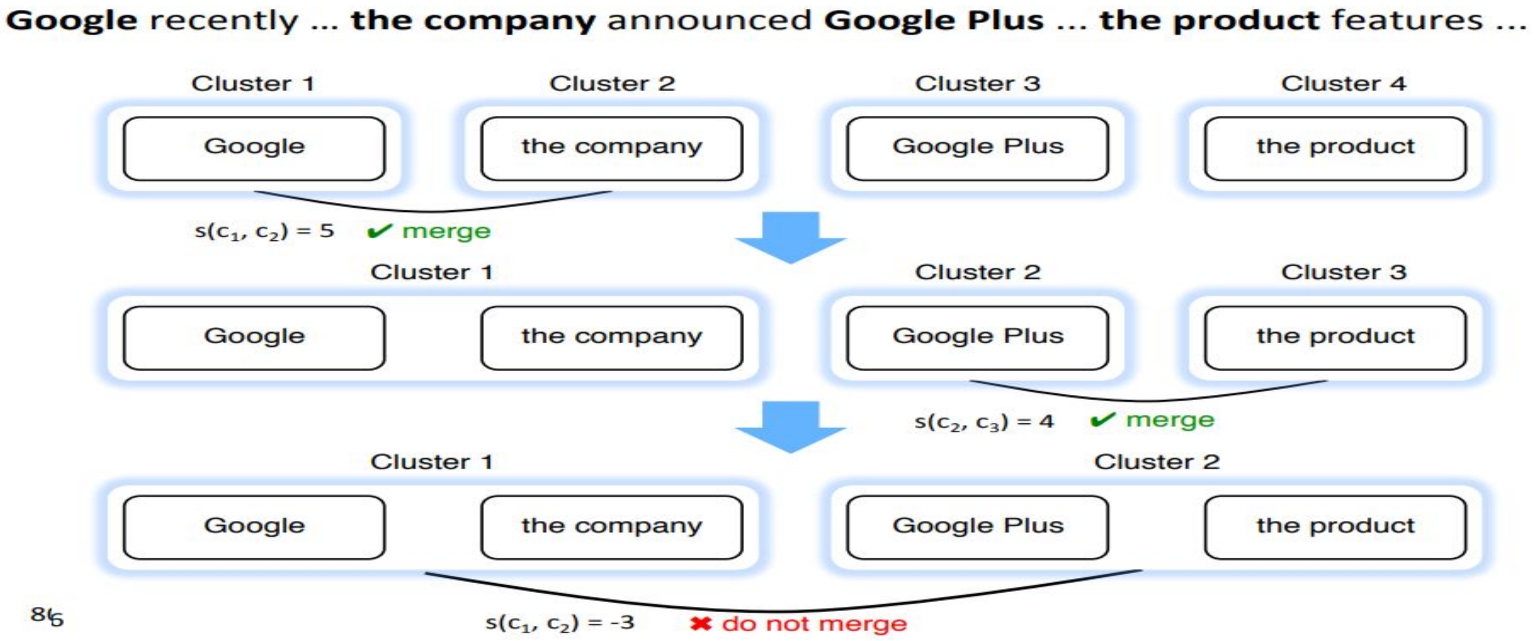
3. Coreference Resolution Clustering Models

Mohit Rajpal

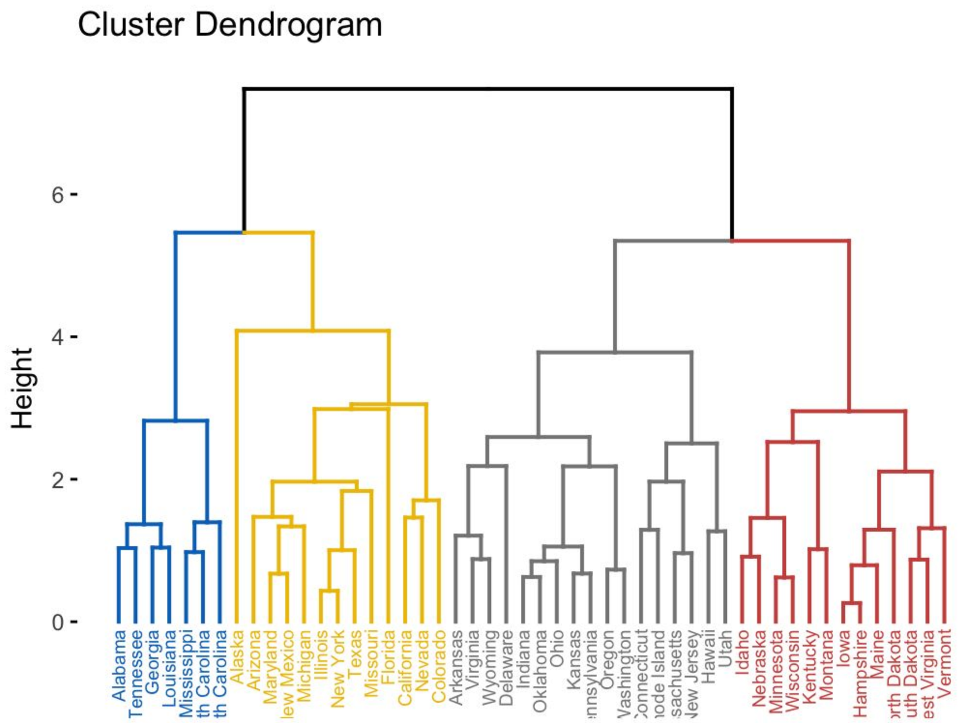
Apologies for the boring white slides

Why (agglomerative) clustering?

- Coreferences have more rich and diverse structure than one-to-one



Agglomerative clustering



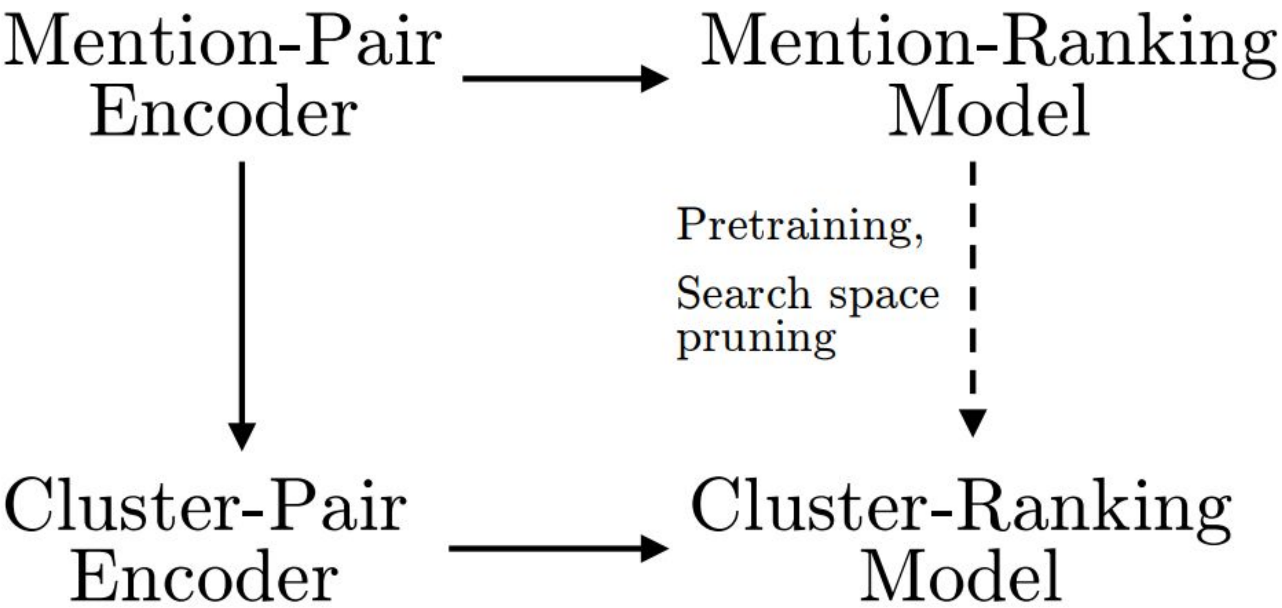
Agglomerative clustering is *really* hard

- How hard? Really hard. No really.
- The hypothesis space is at least as big as all possible binary trees.
- Catalan Numbers: $C_n = \frac{1}{n+1}\binom{2n}{n} = \frac{(2n)!}{(n+1)!\,n!} = \prod_{k=2}^{n} \frac{n+k}{k} \qquad \text{for } n \geq 0.$
- $O(n!) = O(n^n) = O(2^{2^n})$
- EXPSPACE!
- Good news is: you can probably get a theoretical Computer Scientist interested in it because it’s *really* hard.

What I will cover

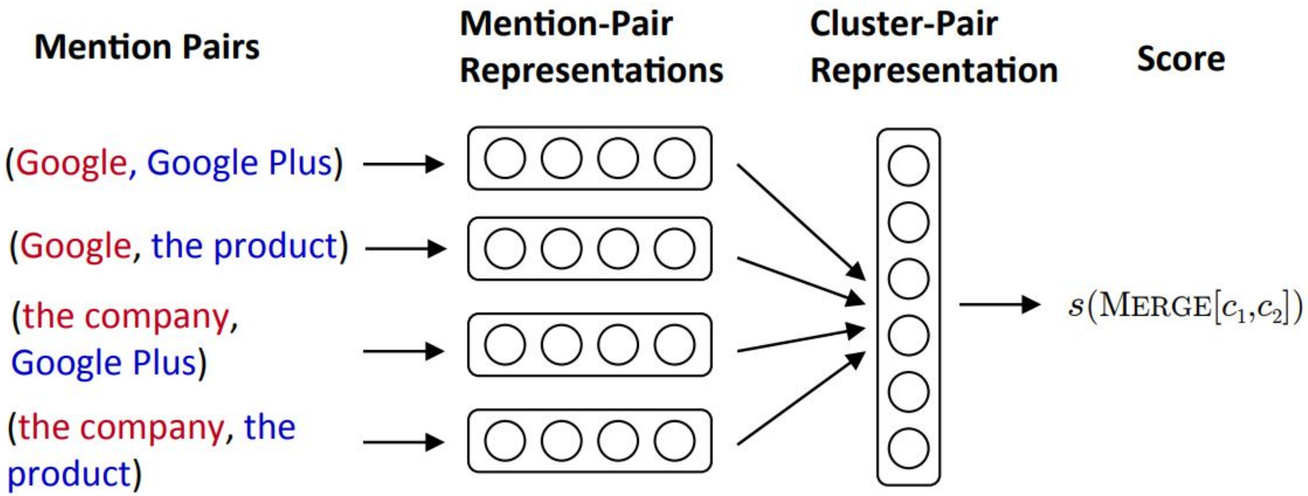
- Problem decomposition
- Neural Network architecture
- Loss functions
- Some (small) comments on feature selection
- Errata

Clustering neural architecture

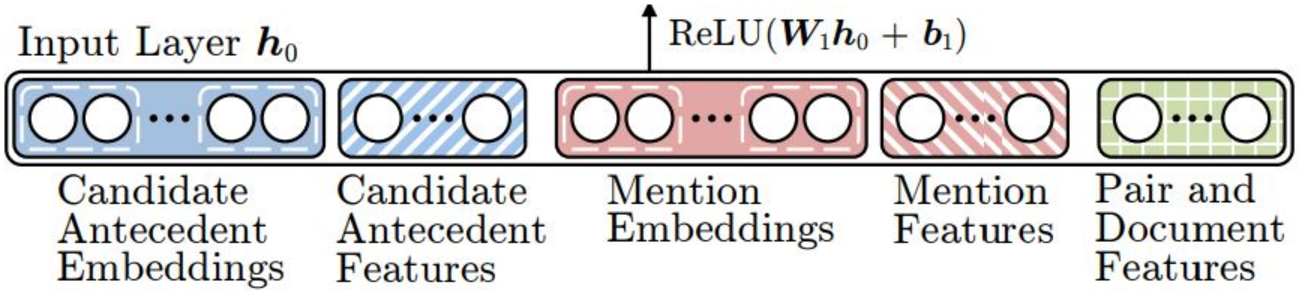


Clustering neural architecture

Merge clusters c_1 = {Google, the company} and
c_2 = {Google Plus, the product} ?



Mention-Pair Encoder features



- Distance from antecedent to mention
- Proximal, and syntactically related words
- Part of speech
- Lots of other features

Mention-Ranking Model loss structure

$$\text{Training set consists of } N \text{ mentions}$$

$$m_1, m_2, m_3, \ldots, m_n$$

$$\text{Let } \mathcal{A}(m_i) \text{ denote the set of candidate andecedents of a mention } m_i$$

$$\text{Let } \mathcal{T}(m_i) \text{ denote the set of true andecedents of a mention } m_i$$

Mention-Ranking Model loss function

m_i:

$$\hat{t}_i = \underset{t \in \mathcal{T}(m_i)}{\operatorname{argmax}}\, s_m(t, m_i)$$

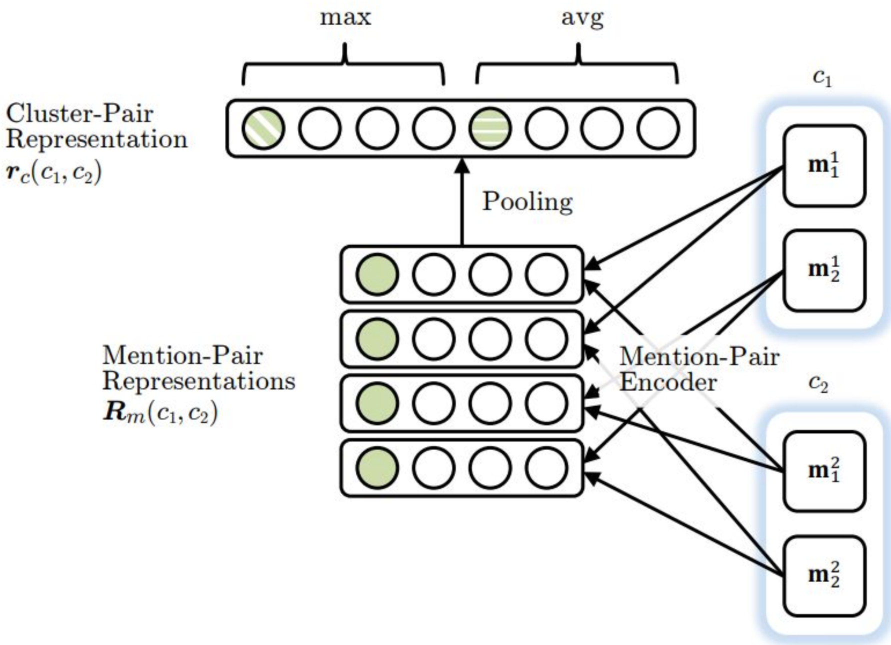
Then the loss is given by

$$\sum_{i=1}^{N} \max_{a \in \mathcal{A}(m_i)} \Delta(a, m_i)(1 + s_m(a, m_i) - s_m(\hat{t}_i, m_i))$$

where $\Delta(a, m_i)$ is the mistake-specific cost function

$$\Delta(a, m_i) = \begin{cases} \alpha_{\text{FN}} & \text{if } a = \text{NA} \wedge \mathcal{T}(m_i) \neq \{\text{NA}\} \\ \alpha_{\text{FA}} & \text{if } a \neq \text{NA} \wedge \mathcal{T}(m_i) = \{\text{NA}\} \\ \alpha_{\text{WL}} & \text{if } a \neq \text{NA} \wedge a \notin \mathcal{T}(m_i) \\ 0 & \text{if } a \in \mathcal{T}(m_i) \end{cases}$$

Cluster-Pair Encoder



Cluster-Ranking Policy network

- Available actions:
  - MERGE$[c_m, c]$, where c is a cluster containing a mention in $\mathcal{A}(m)$. This combines c_m and c into a single coreference cluster.
  - PASS. This leaves the clustering unchanged.

Deep Learning to Search

Algorithm 1 Deep Learning to Search

```
for i = 1 to num_epochs do
    Initialize the current training set Γ = ∅
    for each example (x, y) ∈ D do
        Run the policy π to completion from start state x to obtain a trajectory of states {x_1, x_2, ..., x_n}
        for each state x_i in the trajectory do
            for each possible action u ∈ U(x_i) do
                Execute u on x_i and then run the reference policy π^ref until reaching an end state e
                Assign u a cost by computing the loss on the end state: l(u) = L(e, y)
            end for
            Add the state x_i and associated costs l to Γ
        end for
    end for
    Update π with gradient descent, minimizing Σ_{(x,l)∈Γ} Σ_{u∈U(x)} π(u|x) l(u)
end for
```

Errata

- So we’ve made good progress at hierarchical clustering using NN
- Solving an EXPSPACE problem in PTIME
- Either Neural Networks are a “silver bullet”
- Or coreference resolution is easy
- Option 3?

References

- Stanford CS224n Lecture 13 slides
- Improving coreference resolution by learning entity-level distributed representations by Kevin Clark et. al.