

NLP Week 14 Coreference Resolution

INDEX

1. Introduction

- What is Coreference Resolution
- Application
- Coreference Resolution in two steps

2. Mention Detection

- Three kinds of mentions
- Paradigm shift in Coreference Resolution
- Linguistics of Coreference

3. Linguistic of Coreference

- Anaphora
- Anaphora vs Coreference
- Cataphora

4. Coreference Models

- Rule-based Model - Hobb's naive algorithm
- Mention Pair Model
- Mention Ranking Model
- Mention Ranking Model - Computing probability
- Clustering Model

5. Evaluation & Conclusion

- Coreference Evaluation
- Conclusion

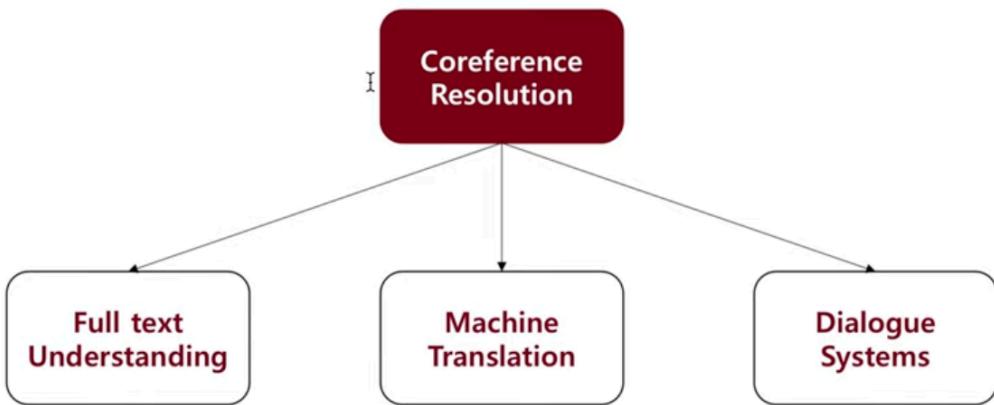
1. Introduction

- **Coreference Resolution :** 상호 참조 해결 → 동일 지시어 찾기!

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.

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 → 여러 문제를 푸는데 적용가능하며, 성능 향상에 도움이 됨



Mention : 개체를 지시하고 있는 텍스트의 범위

mention을 찾는 detection을 먼저 수행하고 같은 mention끼리 clustering 진행



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

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

2. Mention Detection

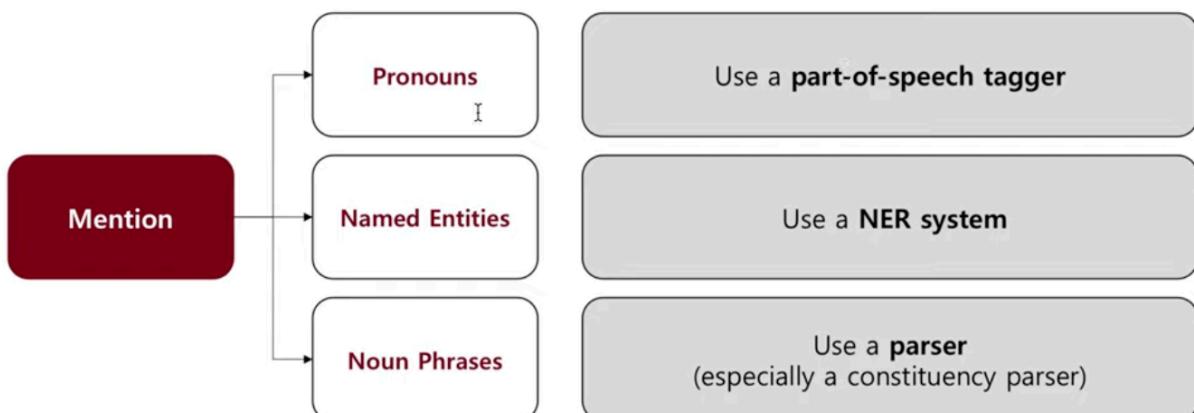
Mention 종류 3가지

pronouns(I,your..)

Named Entities (people, place)

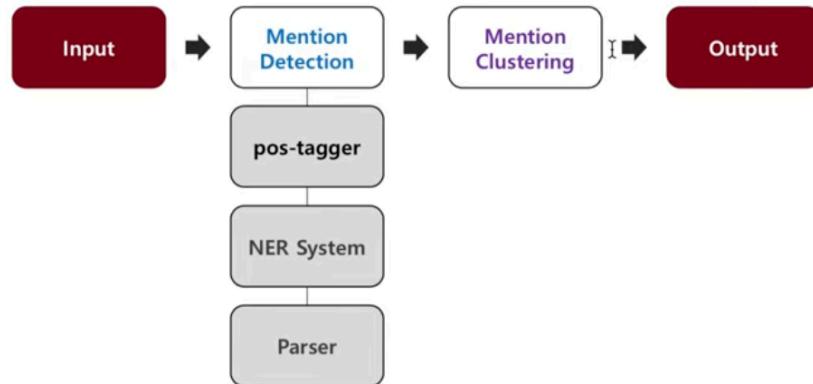
Noun phrase ("a dog")

멘션 찾는 방법 - 기존 자연어 처리



2. Mention Detection

- 2016년 이전까지 대부분, 아래와 같은 2-step 의 파이프라인 시스템을 사용했음



- Pipeline → End-to-End



3. Linguistics of Coreference

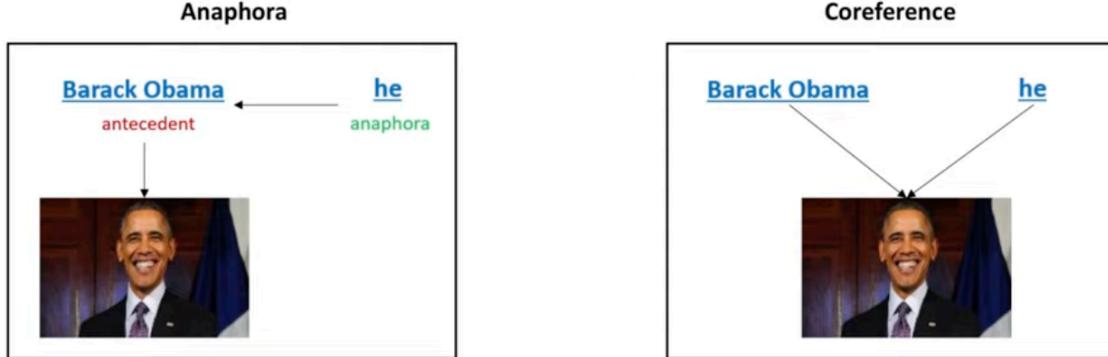
Anaphora : 전방조응 선행어구의 대용으로서 대명사 등을 쓰는 일
문장 내의 어떤 단어들은 독자적으로 참조 할 수 없다

Barack Obama said he would sign the bill.

antecedent anaphora

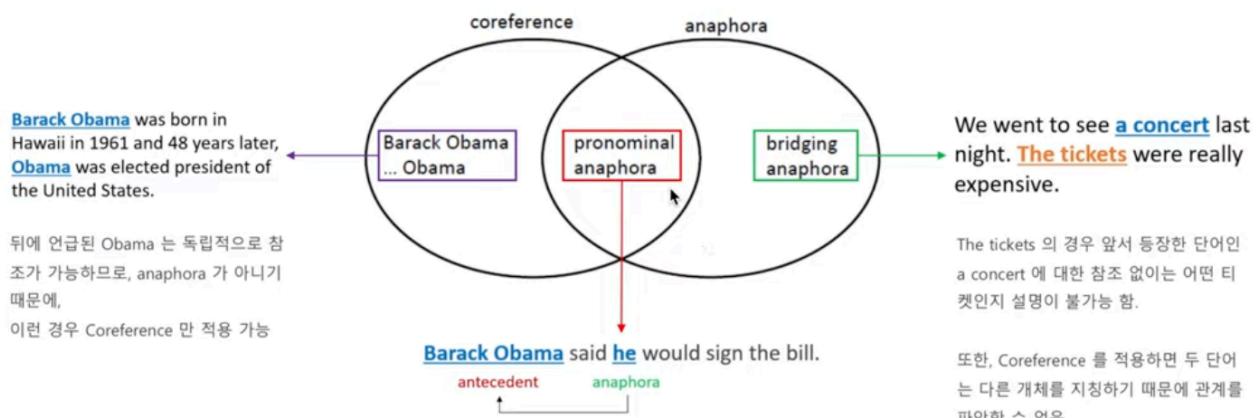
- Anaphora vs. Coreference

Barack Obama said he would sign the bill.



Anaphora : He -> BO -> BO
Coreference He & BO -> BO

- Anaphora vs. Coreference



BO = O

A concert =/= The tickets

- Cataphora

- [문법] 후방 대응(照應), 후속 어구를 지시하는 어구의 사용.
- 대부분의 antecedent 는 anaphora 앞에 등장하지만 항상 그런 것은 아니다.

After he had received his orders, the soldier left the barracks.

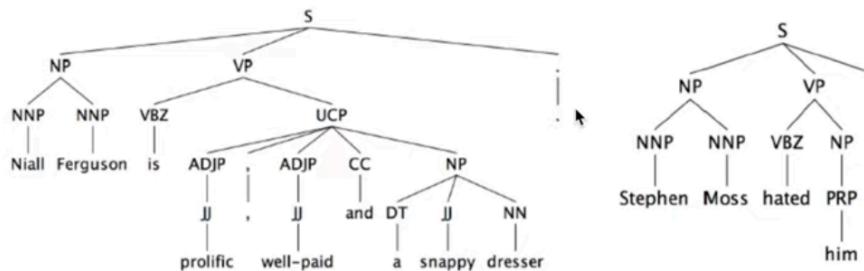


4. Coreference Models

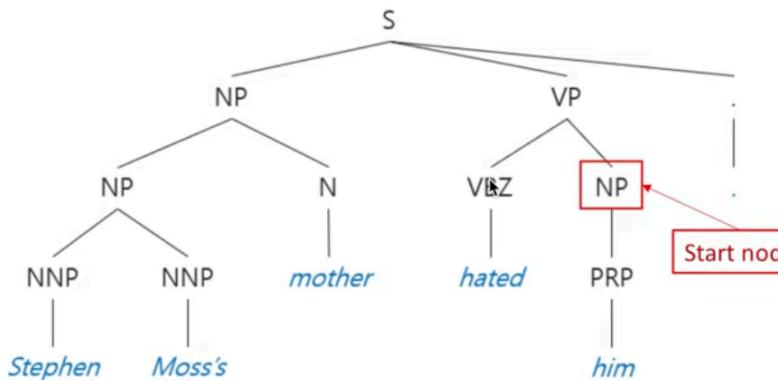
- 1) Rule based Model - hobb's naive algorithm
- 2) Mention pair model
- 3) mention ranking model
- 4) clustering model

Hobb's naive algorithm

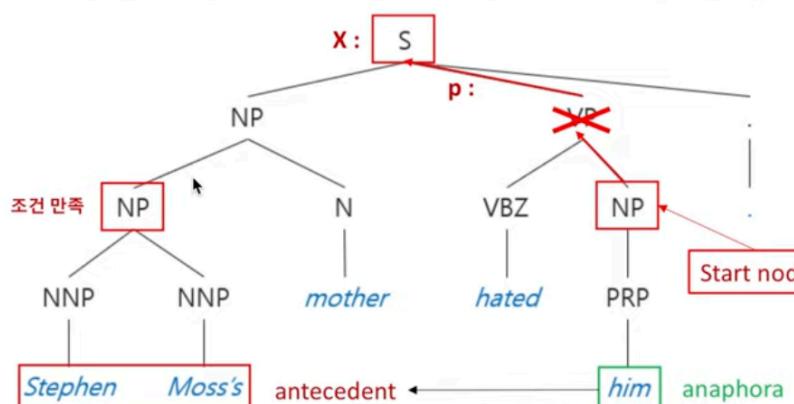
- Hobb's naïve algorithm (1976)
 - Traditional pronominal anaphora resolution
 - 딥러닝 등장 이전에는 ML system 의 feature 로 자주 사용 되었음
 - 총 9개의 step 을 계속해서 반복하면서, antecedent 를 찾아가는 과정을 반복



- Step 1 ~ 3 (+5 ~ 9) : 같은 문장 내에서 선행사를 찾는 과정
 - ① Step 1 : 대명사(PRPs, PRP\$) 를 포함하고 있는 NP 를 시작 노드로 설정



- Step 1 ~ 3 (+5 ~ 9) : 같은 문장 내에서 선행사를 찾는 과정
 - ③ Step 3 : X 에서부터 아래, p 보다 왼쪽에 존재하는 모든 branch 들을 탐색 (left-to-right, breadth-first)
 - 탐색 과정에서 X 와 자기 자신 사이에 NP 또는 S 를 부모로 하는 NP 노드를 찾으면 선행사로 판단
 - ✓ Step 5 ~ 9는 아래 예시와 달리 한 번에 가장 상위 S 노드 까지 올라오지 못 했을 때 하위 노드에서 선행사를 찾는 과정 (생략)



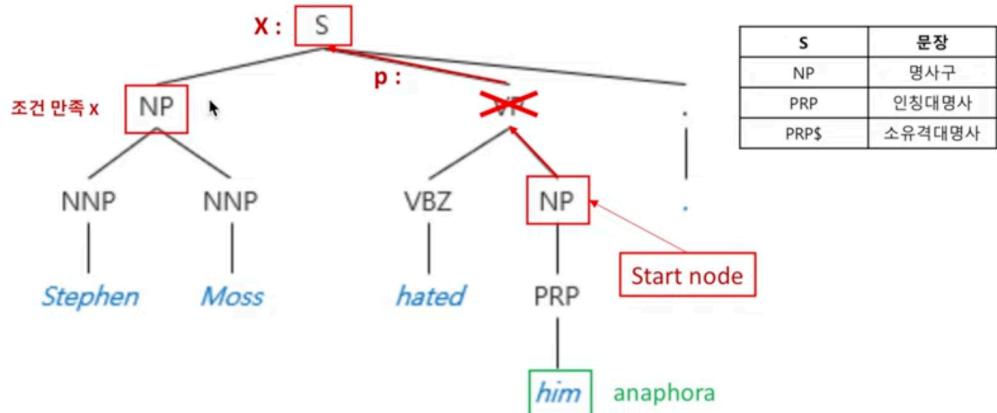
- Step 1 ~ 3 (+5 ~ 9) : 같은 문장 내에서 선행사를 찾는 과정



③ Step 3 : X 에서부터 아래, p 보다 왼쪽에 존재하는 모든 branch 들을 탐색 (left-to-right, breadth-first)

탐색 과정에서 X 와 자기 자신 사이에 NP 또는 S 를 부모로 하는 NP 노드를 찾으면 선행사로 판단

✓ 아래와 같이 해당 문장 내에서 선행사가 없는 경우도 있음, 이럴 경우 이전 문장에서 선행사를 찾는 과정을 같이 수행 → Step 4

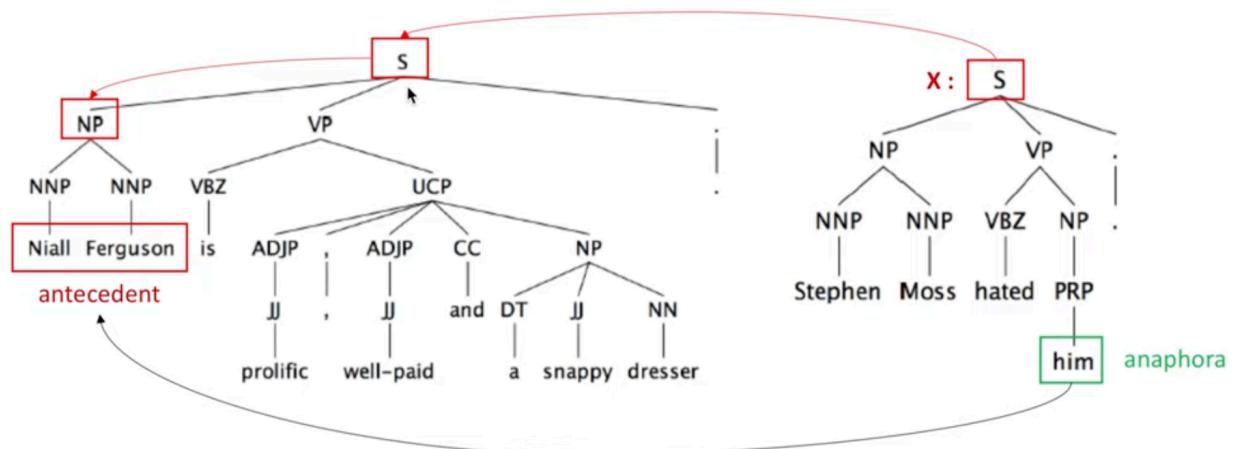


- Step 4 : 이전 문장에서 선행사를 찾는 과정



④ Step 4 : X 가 가장 상위 노드일 때, 이전 문장으로 이동하여 Tree 의 모든 branch 들을 탐색 (left-to-right, breadth-first)

NP 노드를 찾으면 해당 노드를 선행사로 선정. (X 가 최상위 노드가 아니면, step 5 로 이동)

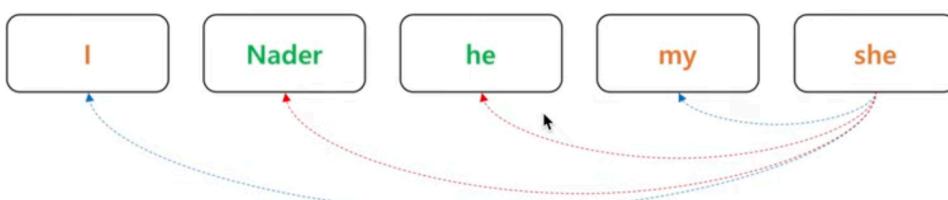


Mention pair model

- Mention Pair Model

▪ 두 개의 mention 간에 coreference 가 있는지 없는지 판단하는 classifier 를 훈련시킴

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



Positive examples → $p(m_i, m_j)$ 가 1에 가까워 지도록 학습

Negative examples → $p(m_i, m_j)$ 가 0에 가까워 지도록 학습

가능한 mention 조합들의 cross-entropy loss를 줄이는 방향으로 학습

i : mention, j : antecedent 의미한다.

i는 2부터 시작 : i=1이면 참조할 단어 없음

Mention Pair Model - Test time

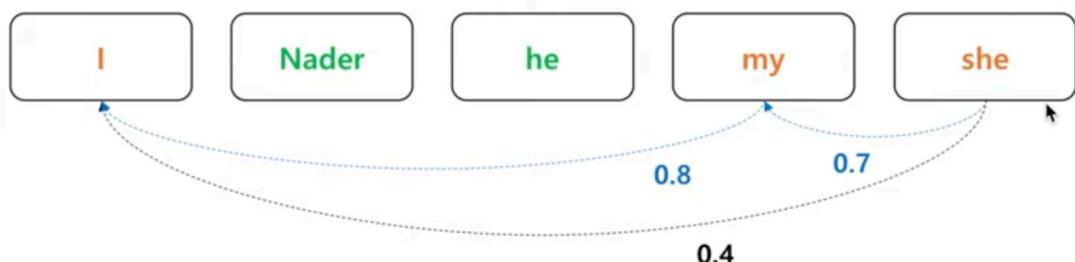
Coreference resolution은 clustering task

하지만 앞서 학습한 모델은 두 mention간 coreference가 있는지만 판단가능->

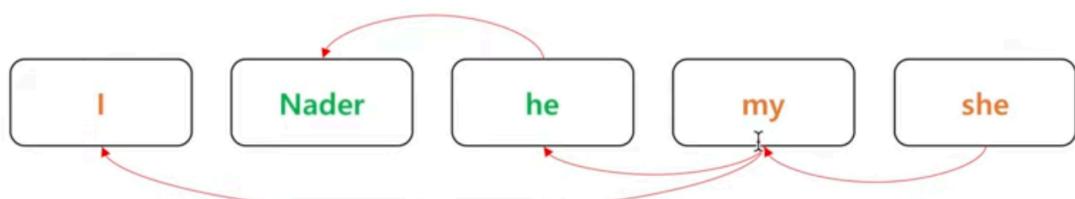
Mention들간의 coreference link는 판단 불가

Clustering을 수행하기 위해, 적절한 threshold 를 넘는 mention끼리 coreference link 추가

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



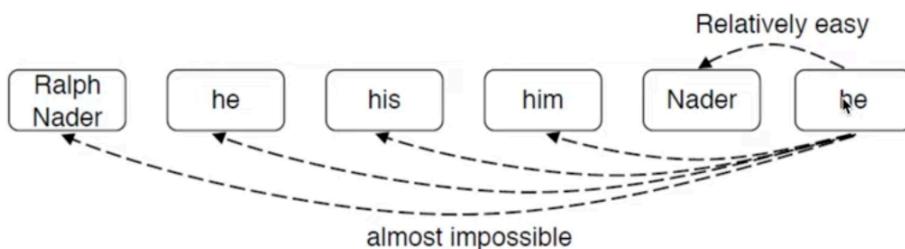
"I voted for Nader because he was most aligned with my values," she said.



반면에 위와 같이 잘못 link 가 이어질 경우 매우 큰 Cluster 가 생기며, 안 좋은 결과가 나올 수도 있음!

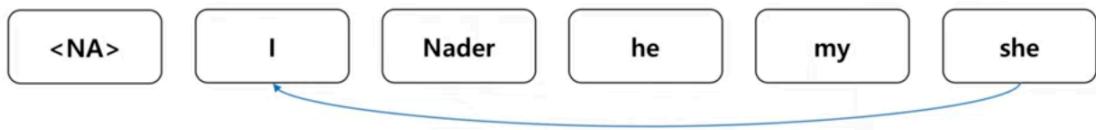
Mention Pair Model - Disadvantage

- 매우 긴 문장이 있다고 가정할 때, 상대적으로 거리가 먼 단어가 선행사일 가능성은 낮음
- 대부분의 mention들은 한 개의 분명한 선행사를 일반적으로 가지고 있음
- 하지만, Mention Pair 모델은 앞서 나온 모든 단어에 대해서 coreference 를 계산하는 방식
- 이런 단점을 해결하기 위해, 한 개의 best 선행사를 찾는 방식으로 모델을 훈련 시켜보자! → Mention Ranking Model



- Mention Ranking Model

- 가장 선행사일 가능성이 높은 mention 을 찾기 위해 model 의 score 에 softmax 연산을 수행
- 선행사가 없을 수 있는 mention 에 대해 처리하기 위해 dummy mention 인 <NA> 를 문장 가장 앞에 추가
- 가장 확률이 높은 한 개의 mention 만을 coreference link 에 추가



$$P(\text{NA}, \text{she}) = 0.1$$

$$P(\text{I}, \text{she}) = 0.5$$

$$P(\text{Nader}, \text{she}) = 0.1$$

$$P(\text{he}, \text{she}) = 0.1$$

$$P(\text{my}, \text{she}) = 0.2$$

- Mention Ranking Model - Training

- 현재 mention 인 m_j 가 선행사 후보 mention 들 중 어떤 하나와 연결될 확률을 최대화 할 수 있는 모델을 원함
- Softmax 를 취하기 전, 확률 값들의 sum 이기 때문에 값이 커질 수 있음 $\rightarrow \log$

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

Iterate through candidate antecedents (previously occurring mentions) For ones that are coreferent to m_jwe want the model to assign a high probability

확률값 최대화하는 모델

- Mention Ranking Model - Training

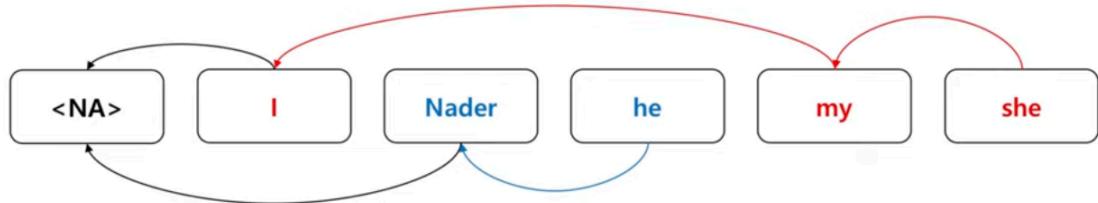
- Softmax 를 취하기 전, 확률 값들의 sum 이기 때문에 값이 커질 수 있음 $\rightarrow \log$
- 최종적으로, 아래의 목적 함수를 최소화 시키는 방향으로 모델 학습 진행

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

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

- Mention Ranking Model – Test Time

- 각각의 mention에 대해서 softmax 값이 가장 높은 하나의 선행사만을 연결시켜 줌
- 하나의 선행사만을 연결시켜 주었음에도, mention-pair 모델과 거의 똑같은 결과를 보여줌
- 별도의 clustering 과정이 없었지만, coreference 가 있는 단어끼리 잘 묶어줄 수 있음



확률 구하는 방법

- Statistical classifier
- Simple neural network
- More advanced model using LSTM, attention

Coreference는 기본적으로 clustering task! -> Clustering 알고리즘

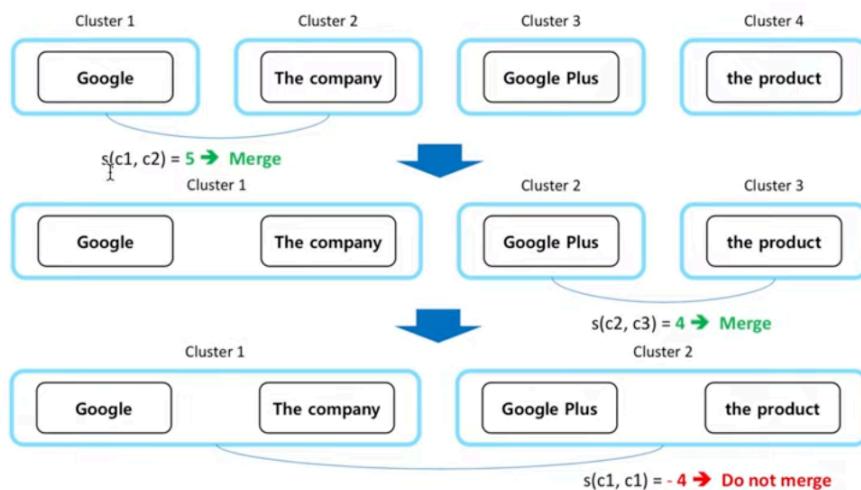
계층적 군집화 방식 사용

- 각각의 mention이 독자적인 cluster를 가지고 있는 상황에서 시작
- 매스텝마다 두개의 cluster씩 merge
- 이때 두개의 cluster를 merge하는 것이 좋은지 아닌지 모델이 score를 매길수 있도록 구성

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



- Clustering process overview



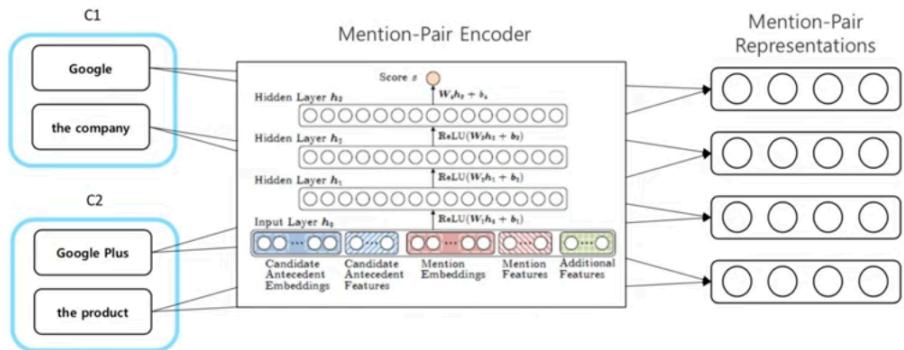
Clustering model architecture

모델 구성은 mention paired를 rep을 cluster pair rep으로 변형하여 Score를 매기게 된다.

- Clustering Model Architecture

- ① Mention-Pair Representations 생성

- 앞서 살펴보았던, Feed-Forward Neural Network 기반의 Mention-Pair Encoder 모델을 사용하여 생성



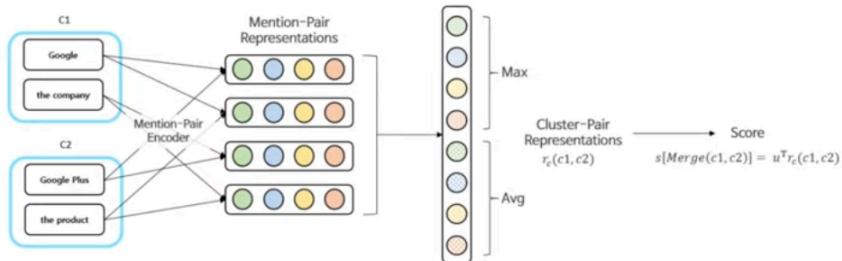
- Clustering Model Architecture

- ② Cluster-Pair Representation 생성

- Pooling operation (\max , avg)을 사용하여 Cluster-pair representation 을 생성

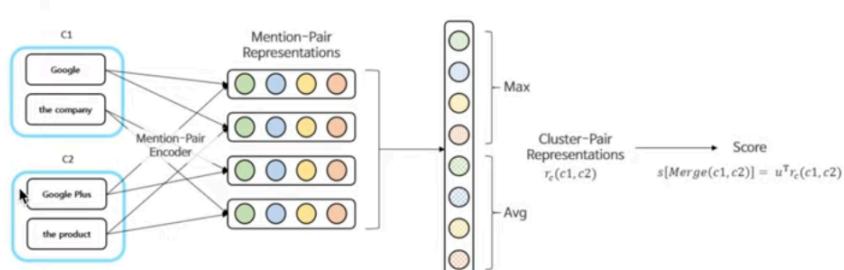
- ③ Score 산출

- Cluster-Pair representation 을 다시 한번 Linear transform 하여 최종 score 산출



- Clustering Model Training

- 현재 Merge 할 Cluster Pair 는 이전 step 에 이미 만들어진 cluster 들에 의존적임
- 일반적인 Supervised Learning 으로 풀 수 없음 → 강화학습 사용
- 매 스텝마다 merge 결과에 따른 Coreference Evaluation Metric 의 변화를 Reward 로 사용



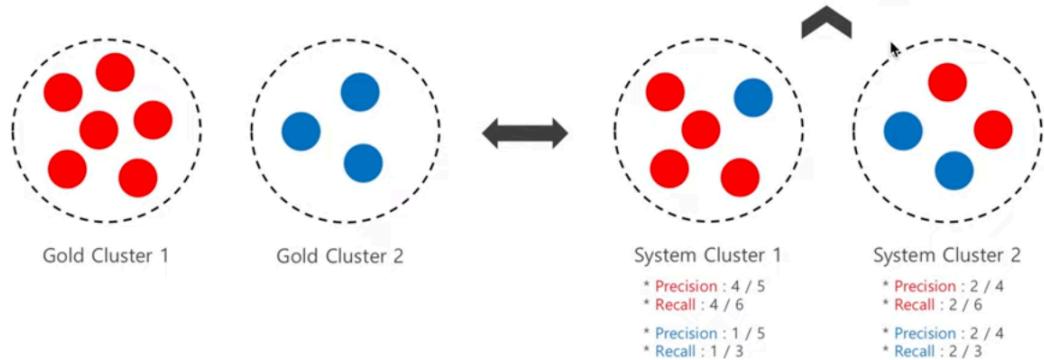
5. Evaluation & conclusion

- Many different metrics : MUC, CEAFL, LEA, B-CUBED, BLANC

- 보통, 여러 metric 들의 평균 값으로 성능을 판단
- 이번 강의에서는, B-CUBED metric 만을 소개 하였음
- B-CUBED : 개별적으로 Precision, Recall 을 구한 뒤, Average

$$\text{Precision} : [4*(4/5) + 1*(1/5) + 2*(2/4) + 2*(2/4)] / 9 = 0.6$$

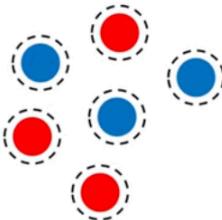
$$\text{Recall} : [4*(4/6) + 1*(1/3) + 2*(2/6) + 2*(2/3)] / 9 = 0.56$$



- Find a balance point

- Under Cluster → Precision 은 높아지지만, Recall 이 줄어듬
- Over Cluster → Recall 은 높아지지만, Precision 이 줄어듬
- 균형 있는 지점을 찾는 것이 중요!

Precision : 1.0 / Recall : 0.33



Precision : 0.5 / Recall : 1.0

