# Lecture 15: Coreference Resolution

2018 年 8 月 5 日

# 1 Introduction

## 1.1 Coreference

In linguistics, coreference, sometimes written co-reference, occurs when two or more expressions in a text refer to the same person or thing; they have the same referent.

## 1.2 Types of Coreference

- **Anaphora 回指**

  The $music_i$ was so loud that $it_i$ couldn't be enjoyed.

  The anaphor($it_i$) follows its antecedent($music_i$).

- **Cataphora**

  If $they_i$ are angry about the music, the $neighbors_i$ will call the cops.

  The anaphor($they_i$) precedes its antecedent($neighbors_i$).

- **Split antecedents**

  $Carol_i$ told $Bob_i$ to attend the party. $They_i$ arrived together.

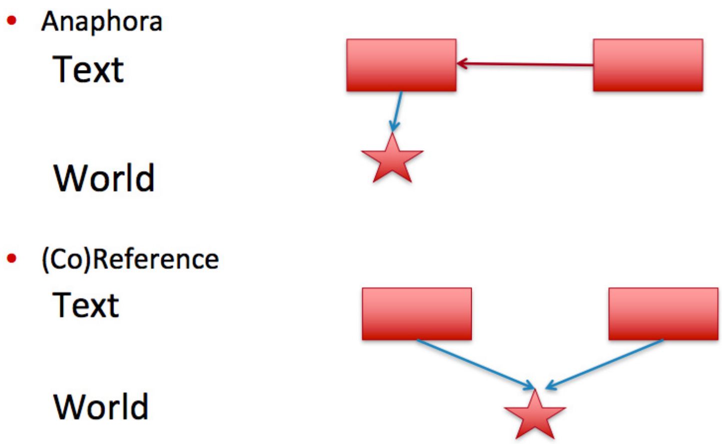
  The anaphor *they* has a split antecedent, referring to both $Carol$ and $Bob$.

- **Coreferring noun phrases 共指**

  $The\ project\ leader_i$ is refusing to help. The $jerk_i$ thinks only of himself.

  Coreferring noun phrases, whereby the second noun phrase is a predication over the first.

在 NLP 领域，一般只研究回指和共指。



## 1.3 Coreference Resolution

Identify all noun phrases that refer. Find which entity in physical world that the noun refer to.

### Mentions

- 人名: Akash, Krishna
- 代词: His, herself, it
- 普通词组: A couple of years, a naughty child

### Applications

- 机器翻译（土耳其语不区分‘他’和‘她’）。
- IR 和 QA: 对代词进行提问。
- 全文理解与文本摘要。

### 几种指代消解模型

- Mention Pair models:

  将所有 anaphor 与 antecedent 视作一系列 pair，对每个 pair 进行二分类。

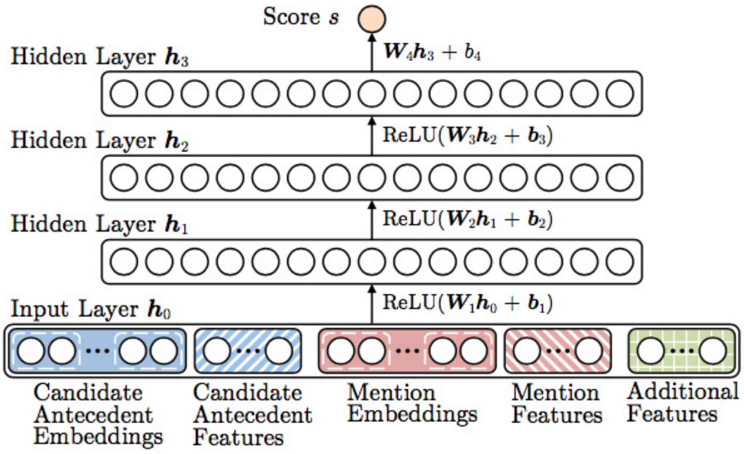
- Mention ranking model:

  显式地将 mention 作为 query，对 candidate antecedents 做 ranking。

- Entity-Mention models

  找出所有 entity 及其上下文，通过上下文进行聚类，在同一类中的 mention 消解为同一个 entity。

## 2 Simple Neural Network Model

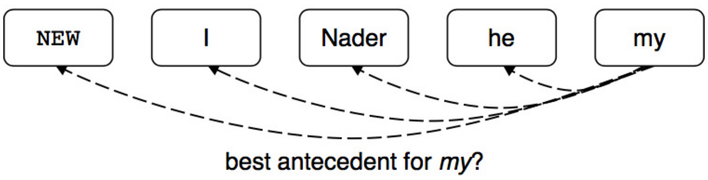


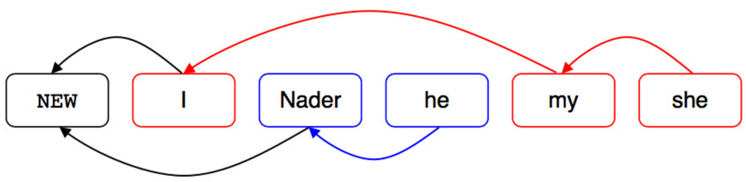

最基本的前馈神经网络，输入特征包括 Current Mention Embedding 以及 Candidate Antecedent Embeddings 加上大量的 hand-crafted Features.

- 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.
- …

实验表明，这些手工 feature 仍然具有非常大的作用，人们在致力于减少这些 features，最新的模型已经将其减少至 13 个。

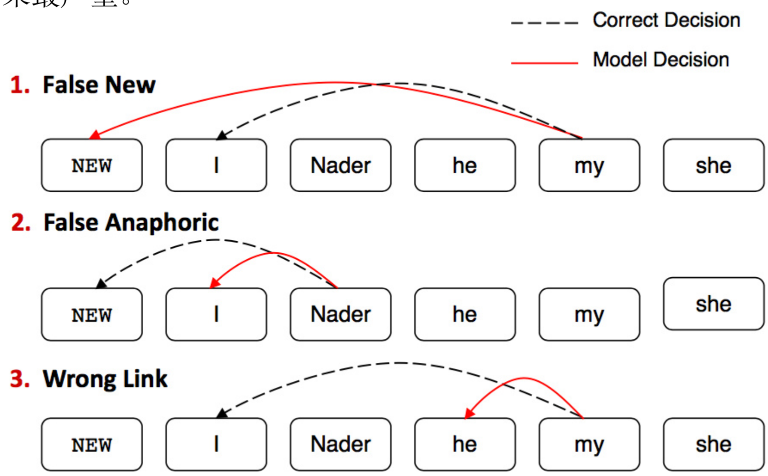
**模型 idea** 对每一个 entity，考虑那些之前的词与之构成指代，对每一个 pair 放进 model 进行打分。新出现的词与 NEW 相连。






# 3 Reinforcement Learning

虽然可以根据分数来优化模型，然而不同预测错误的严重性是不同的，如下图第三种错误的后果最严重。




## 3.1 Rl 算法 1

将 mention-pair 模型的打分 softmax 成概率，最大化决策序列的奖励期望。

- Define probability distribution over actions:
  $p_\theta((c,m)) \propto e^{s(c,m)}$ for any action $a = (c,m)$
- Maximize expected reward
  $J(\theta) = \mathbb{E}_{[a_{1:T} \sim p_\theta]} R(a_{1:T})$
  - Sample trajectories of actions to approximate gradient

缺陷：该算法存在一个很严重的问题，所有决策序列的奖励期望是提高了，但我们真正目的并不在此，而是想要的是让得分最高的那一个决策序列的分值尽量高。

## 3.2 Reward-Rescaling 算法

不再人工指定惩罚系数，而是用当前决策序列下，改动某一个决策所带来的奖励的下降来作为惩罚系数。RL 实际上犯了更多的错误，但它擅长不犯致命错误.