CS11-747 Neural Networks for NLP

Document Level Models

Graham Neubig

Carnegie Mellon University
Language Technologies Institute

Site
https://phontron.com/class/nn4nlp2019/

(w/ thanks for many Slides from Zhengzhong Liu)

Some NLP Tasks we've Handled

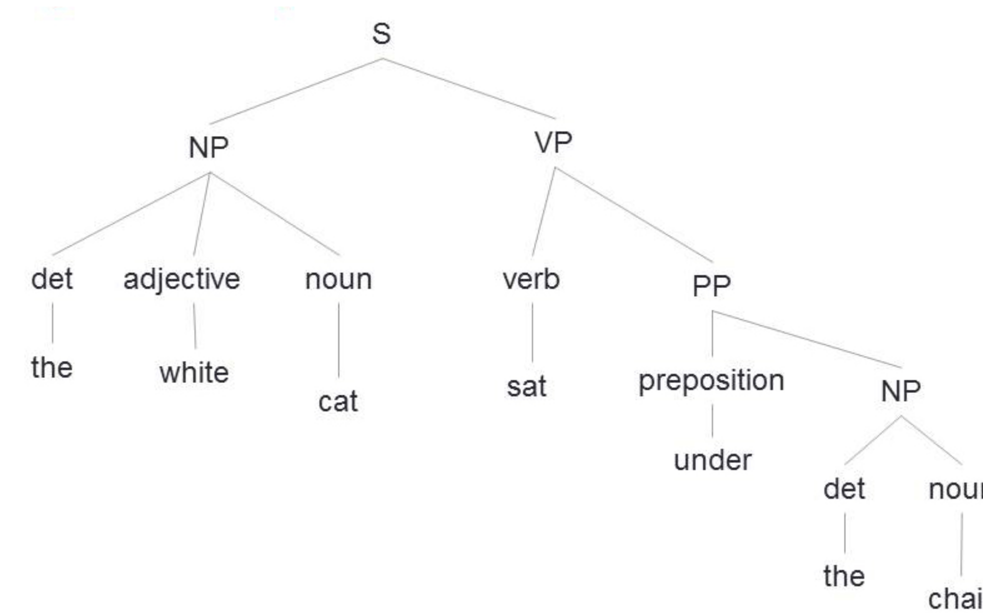
Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, 'and what is the use of a book,' thought Alice 'without pictures or conversation?'

P(w_{i+1}= of | w_i=tired) = 1
P(w_{i+1}= of | w_i=use) = 1
P(w_{i+1}= sister | w_i=her) = 1
P(w_{i+1}= beginning | w_i=was) = 1/2
P(w_{i+1}= reading | w_i=was) = 1/2

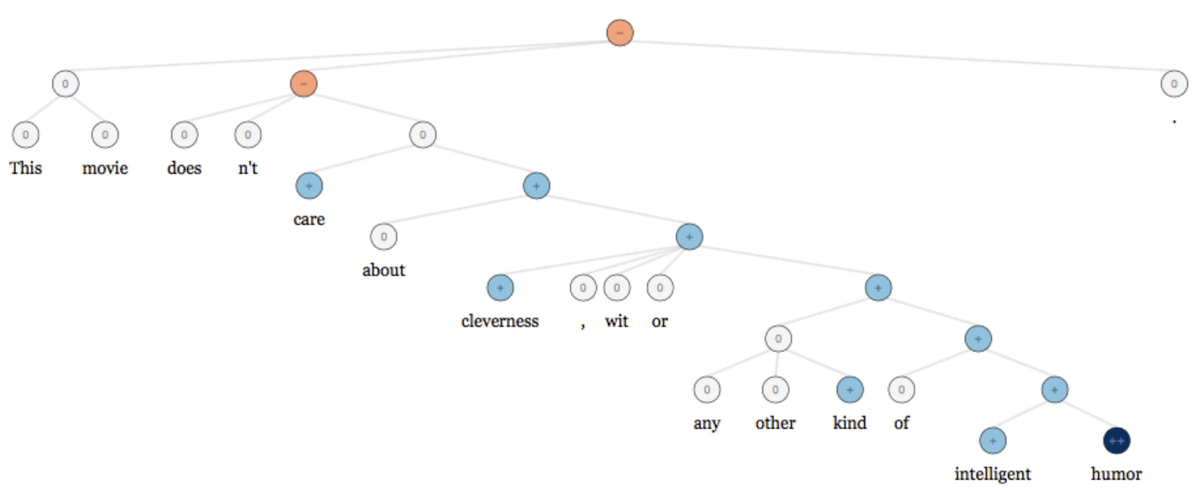
P(w_{i+1}= bank | w_i=the) = 1/3
P(w_{i+1}= book | w_i=the) = 1/3
P(w_{i+1}= use | w_i=the) = 1/3

$$P(X|Y) = \frac{P(X,Y)}{P(Y)}$$

Language Models



Parsing



Classification

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

Entity Tagging

Some Connections to Tasks over Documents

Prediction using documents

- **Document-level language modeling:** Predicting coherence of language on the multi-sentence level (c.f. single-sentence language modeling)

- **Document classification:** Predicting traits of entire documents (c.f. sentence classification)

- **Entity coreference:** Which entities correspond to each-other? (c.f. NER)

- **Discourse parsing:** How do segments of a document correspond to each-other? (c.f. syntactic parsing)

Prediction of document structure

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?'

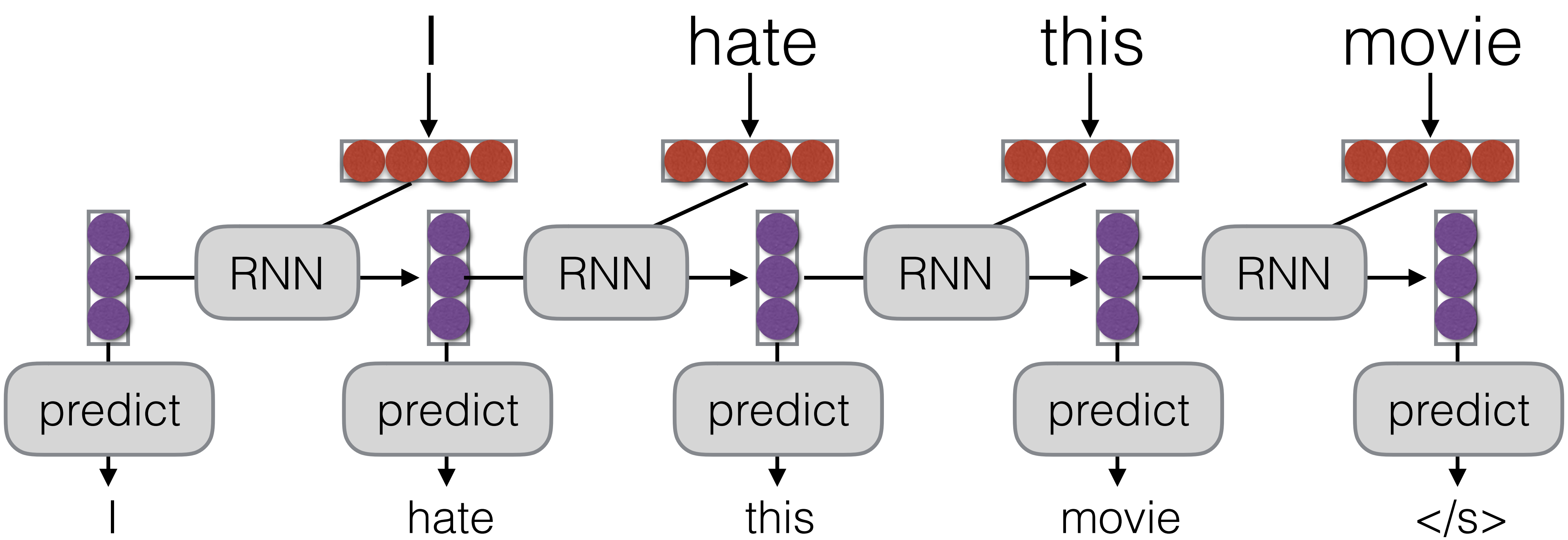
Document Level Language Modeling

Document Level Language Modeling

- We want to predict the probability of words in an entire document
- Obviously sentences in a document don't exist in a vacuum! We want to take advantage of this fact.

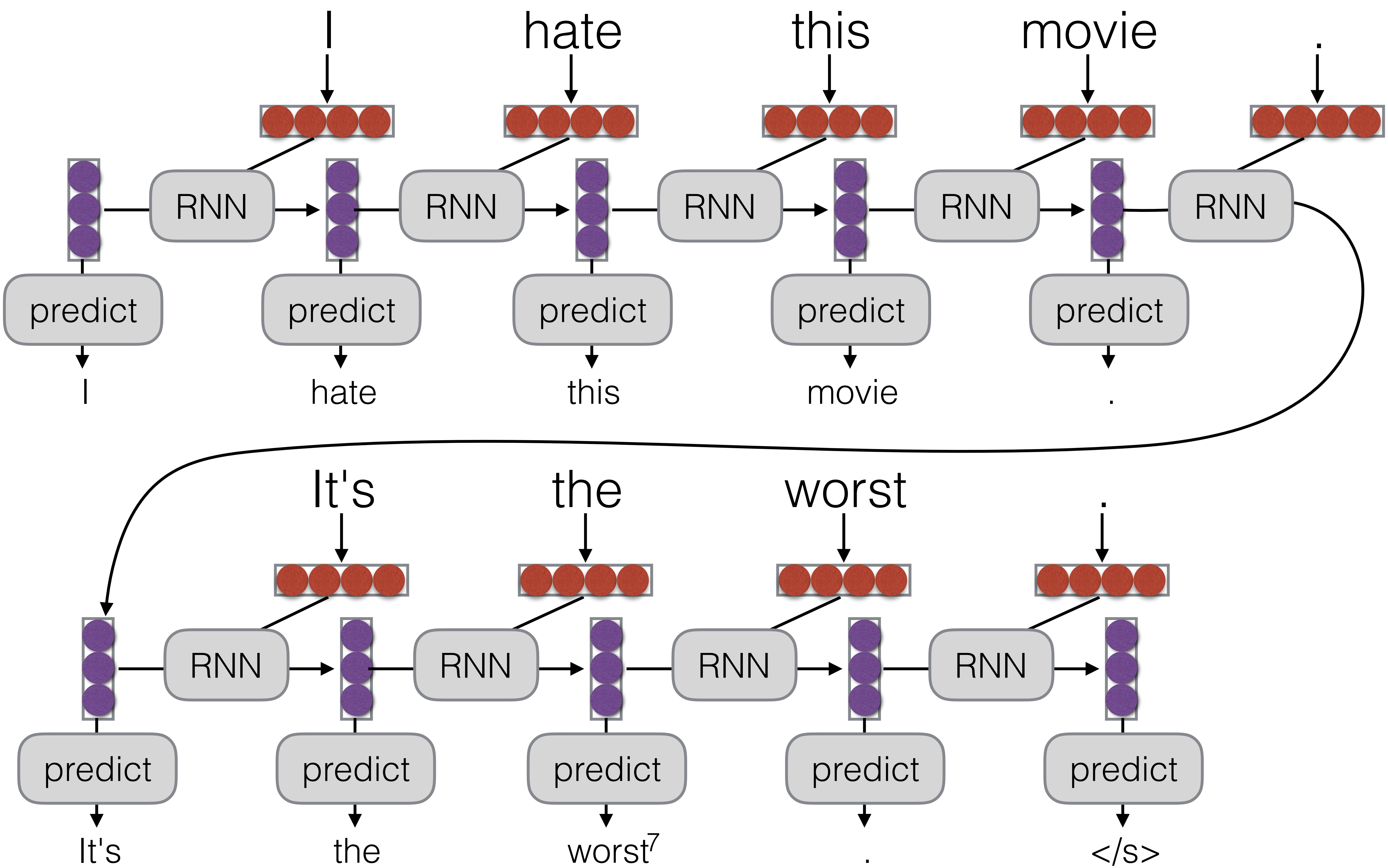
Remember: Modeling using Recurrent Networks

- Model passing previous information in hidden state



Simple: Infinitely Pass State

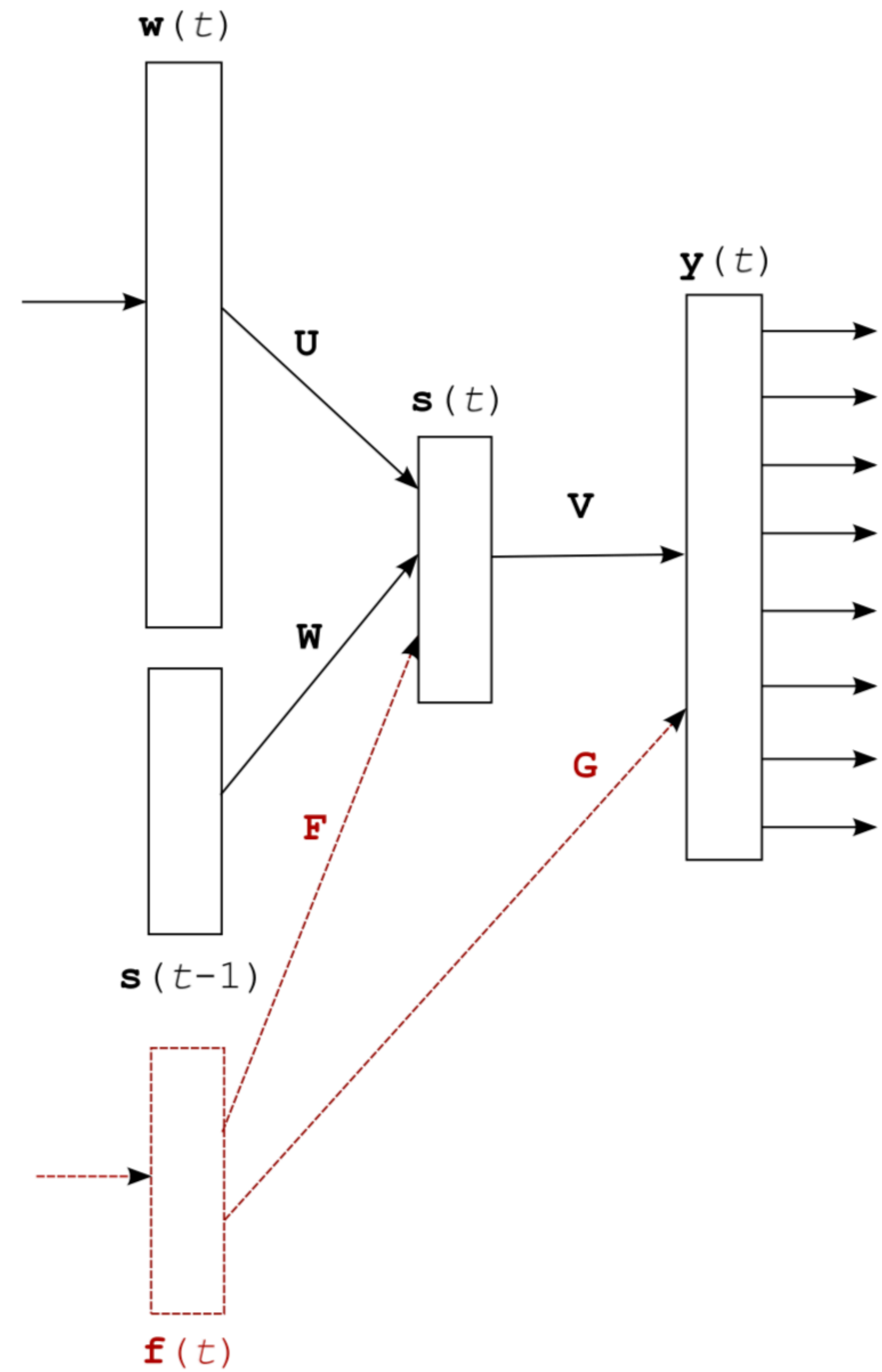
(Mikolov et al. 2011)



Separate Encoding for Coarse-grained Document Context
(Mikolov & Zweig 2012)

- One big LSTM for local and global context tends to miss out on global context (as local context is more predictive)

- Other attempts try to incorporate document-level context explicitly



What Context to Incorporate?

- Use topic modeling (Mikolov and Zweig 2012)
- Use bag-of-words of previous sentence(s), optionally with attention (Wang and Cho 2016)
- Use last state of previous sentence (Ji et al. 2015)

Self-Attention Across Sentences

- Simple idea: attend to the previous sentence (Voita et al. 2018)
    - Concatenate previous sentence tokens with current sentence tokens, attend to all
    - Adds context from previous sentence
- Clever idea: attend to **vectors** from the previous sentence (Dai et al. 2019)
    - Like recurrent self attention
    - Infinite context, but no backprop into previous sentence

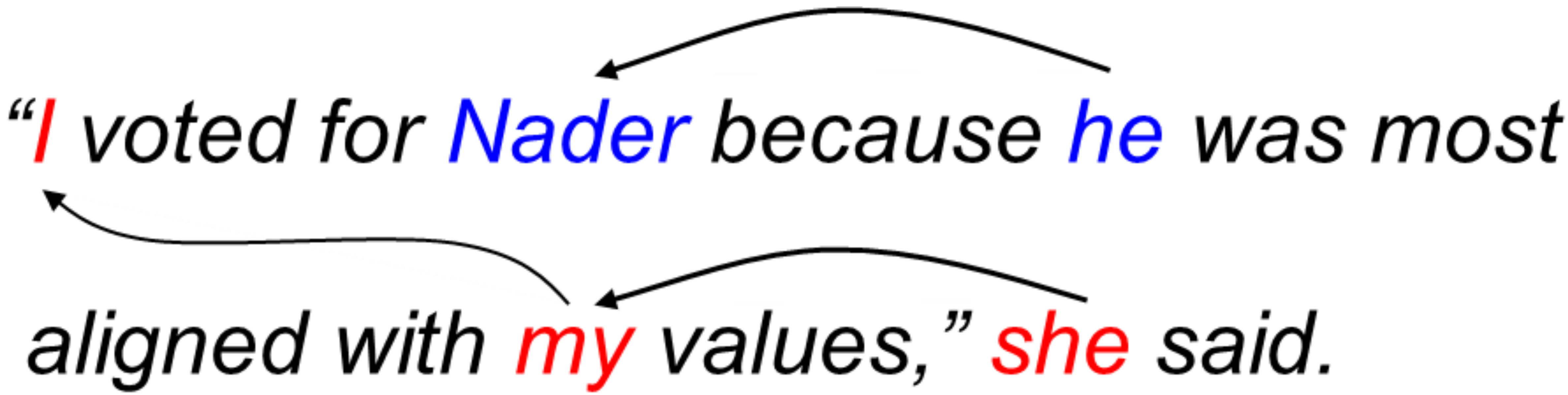
How to Evaluate Document Coherence Models?

- Simple: Perplexity
- More focused:
  - Sentence scrambling (Barzilay and Lapata 2008)
  - Final sentence prediction (Mostafazadeh et al. 2016)

Context	Right Ending	Wrong Ending
Karen 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.	Karen became good friends with her roommate.	Karen hated her roommate.
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.	Jim decided to devise a plan for repayment.	Jim decided to open another credit card.
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.	She found her phone in the car.	She didn't want her phone anymore.

  - Final word prediction (Paperno et al. 2016)

(3) *Context:* Preston had been the last person to wear those <u>chains</u>, and I knew what I'd see and feel if they were slipped onto my skin-the Reaper's unending hatred of me. I'd felt enough of that emotion already in the amphitheater. I didn't want to feel anymore. "Don't put those on me," I whispered. "Please."
Target sentence: Sergei looked at me, surprised by my low, raspy please, but he put down the _____.
Target word: chains



Entity Coreference

Image credit: Stanford NLP

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**.

Mention(Noun Phrase) Detection

A renowned speech therapist was summoned to help the King overcome his *speech impediment*…

A renowned speech therapist was summoned to help the King overcome his *speech impediment*...

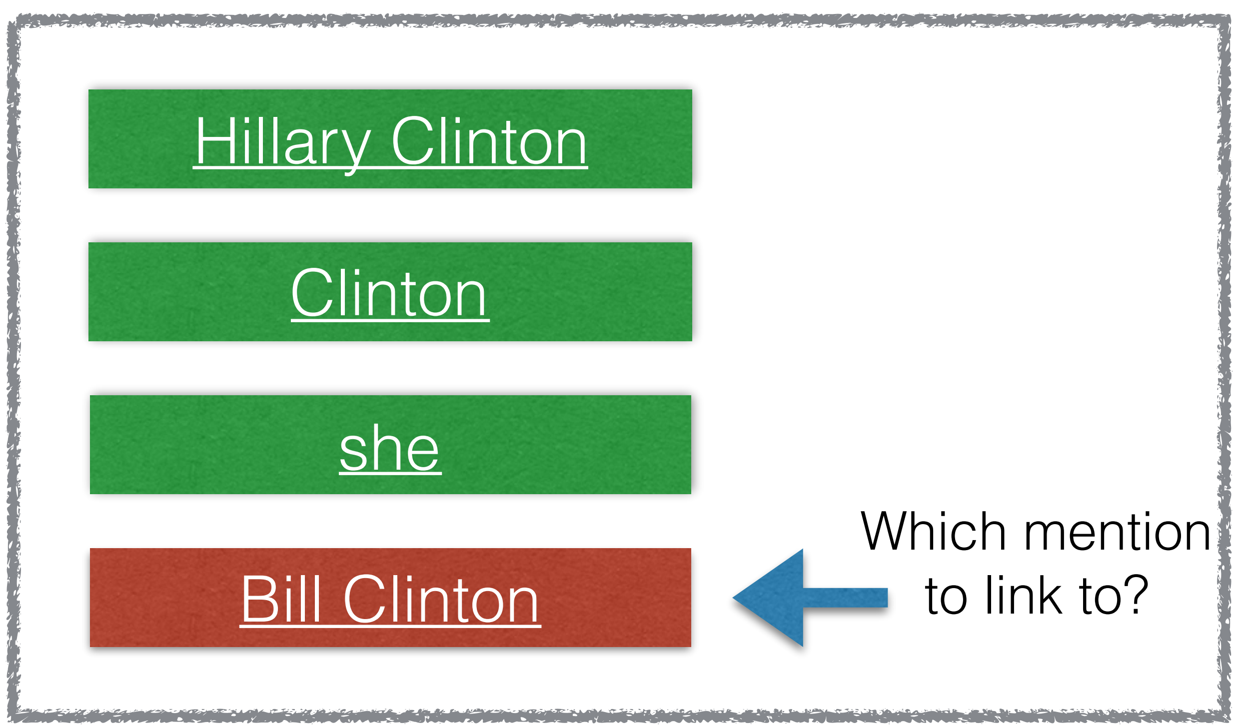
- One may think coreference is simply a clustering problem of given Noun Phrases.
  - Detecting relevant noun phrases is a difficult and important step.
  - Knowing the correct noun phrases affect the result a lot.
  - Normally done as a preprocessing step.

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-based
  - Entity-based



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.

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

✔: Queen Elizabeth <-> her
✘: 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)

* This process often follows the natural discourse order, so we can refer to partially built clusters.

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

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

Entity Models: Entity-Centric Models

Clark and Manning (2015)

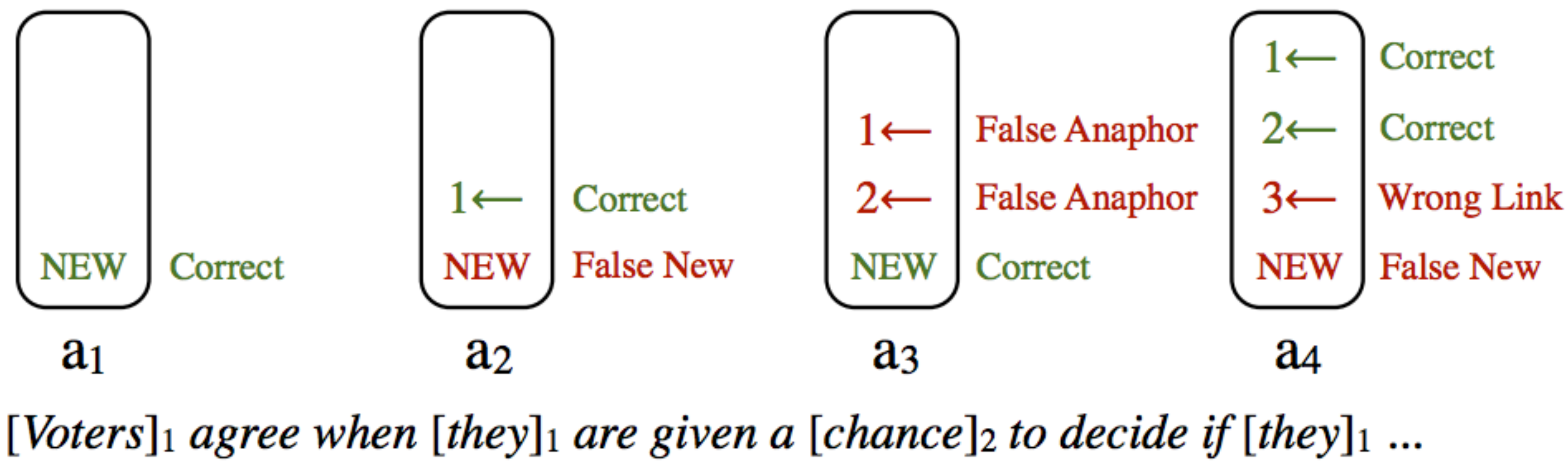
- Entity Centric Models
  - Create an instance between two clusters.
  - Allow building an entity representation.

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 Model: Mention Ranking

(Durrett and Klein, 2013)



A **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)

Features describing m_j, a candidate a		
1	PRONOUN_1	Y if m_j is a pro
2	SUBJECT_1	Y if m_j is a sub
3	NESTED_1	Y if m_j is a nes
Features describing m_k, the mention		
4	NUMBER_2	SINGULAR or PL
5	GENDER_2	MALE, FEMALE common first na
6	PRONOUN_2	Y if m_k is a pro
7	NESTED_2	Y if m_k is a nes
8	SEMCLASS_2	the semantic cl NIZATION, DATI mined using Wo nizer (Finkel, G
9	ANIMACY_2	Y if m_k is deter recognizer; else
10	PRO_TYPE_2	the nominative feature value fo

Features describing the relationship between m_j, a candidate antecedent and m_k, the mention to be resolved (continued from the previous page)		
30	SEMCLASS	C if the mentions have the same semantic class (where the set of semantic classes considered here is enumerated in the description of the SEMCLASS_2 feature); I if they don't; NA if the semantic class information for one or both mentions cannot be determined
31	ALIAS	C if one mention is an abbreviation or an acronym of the other; else I
32	DISTANCE	binned values for sentence distance between the mentions
Additional features describing the relationship between m_j, a candidate antecedent and m_k, the mention to be resolved		
33	NUMBER'	the concatenation of the NUMBER_2 feature values of m_j and m_k. E.g., if m_j is *Clinton* and m_k is *they*, the feature value is SINGULAR-PLURAL, since m_j is singular and m_k is plural
34	GENDER'	the concatenation of the GENDER_2 feature values of m_j and m_k
35	PRONOUN'	the concatenation of the PRONOUN_2 feature values of m_j and m_k
36	NESTED'	the concatenation of the NESTED_2 feature values of m_j and m_k
37	SEMCLASS'	the concatenation of the SEMCLASS_2 feature values of m_j and m_k
38	ANIMACY'	the concatenation of the ANIMACY_2 feature values of m_j and m_k
39	PRO_TYPE'	the concatenation of the PRO_TYPE_2 feature values of m_j and m_k

en m_j, a candidate antecedent and m_k,

e the same head noun; else I
the same string; else I
substring of the other; else I
e pronominal and are the same string; else I
e proper names and are the same string; else I
s are both non-pronominal and are the same

the same modifiers; NA if one of both of them
else I
e pronominal and are either the same pronoun
respect to case; NA if at least one of them is

ree in number; I if they disagree; NA if the
h mentions cannot be determined
e in gender; I if they disagree; NA if the gender
ons cannot be determined
e in both gender and number; I if they disagree
ender; else NA
atch in animacy; I if they don't; NA if the
th mentions cannot be determined
e pronouns; I if neither are pronouns; else NA
e proper nouns; I if neither are proper nouns;

does not have the same maximial NP projec-

pans the other; else I
e NP and is not in an appositive relationship;

in an appositive relationship; else I
in a copular construction; else I

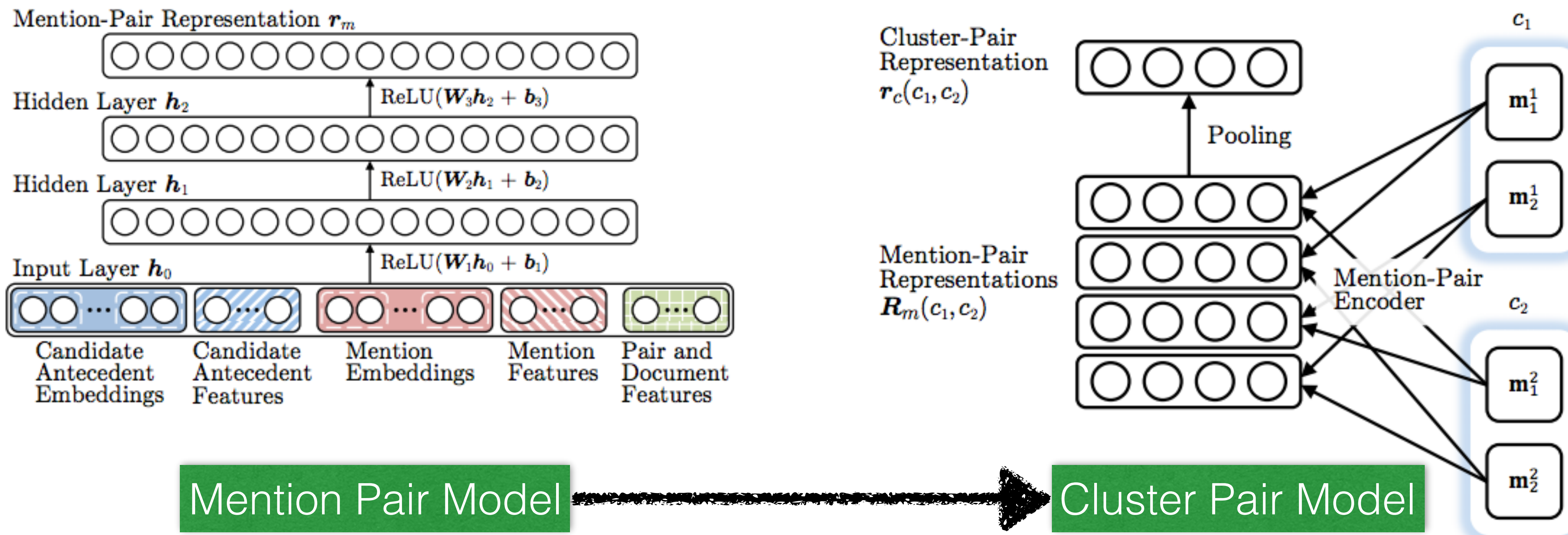
Rank previous clusters for a given mention.
Similarly, a NULL cluster is added to the antecedents.
Rahman & Ng use a complex set of features (39 feature templates)

Advantages of Neural Network Models for Coreference

- **Learn the features** with embeddings since most of them can be captured by surface features.
- **Train towards the metric** using reinforcement learning or margin-based methods.
- **Jointly perform mention detection** and clustering.

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

Feature

- Heuristic Max-Margin as in Wiseman et al.(2015) and Durrett & Klein (2013) *Objective*

- Cluster merging as with Policy Network (MERGE or PASS)
- Trained with SEARN (Daume III et al., 2009)

Training

Deep Reinforcement Learning for Mention-Ranking Coreference Models

Clark & Manning (2016)

- A continuation 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)

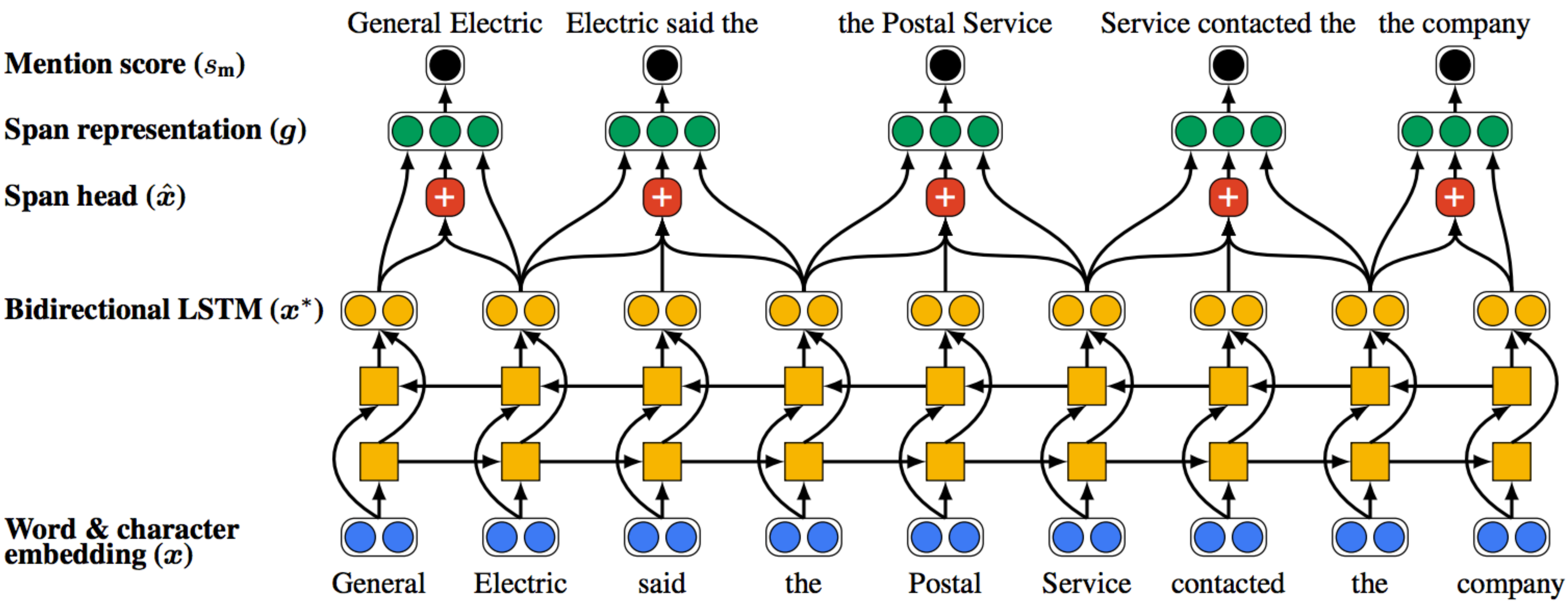
- Cluster level features are difficult to capture.
- Example cluster level features:
  - most-female=true (how to define most?).
  - Pronoun sequence: C-P-P = true.
- Use RNN to embed features from multiple mentions into a single representation.
  - No hand designed cluster level feature templates.

End-to-End Neural Coreference

Lee et.al (2017)

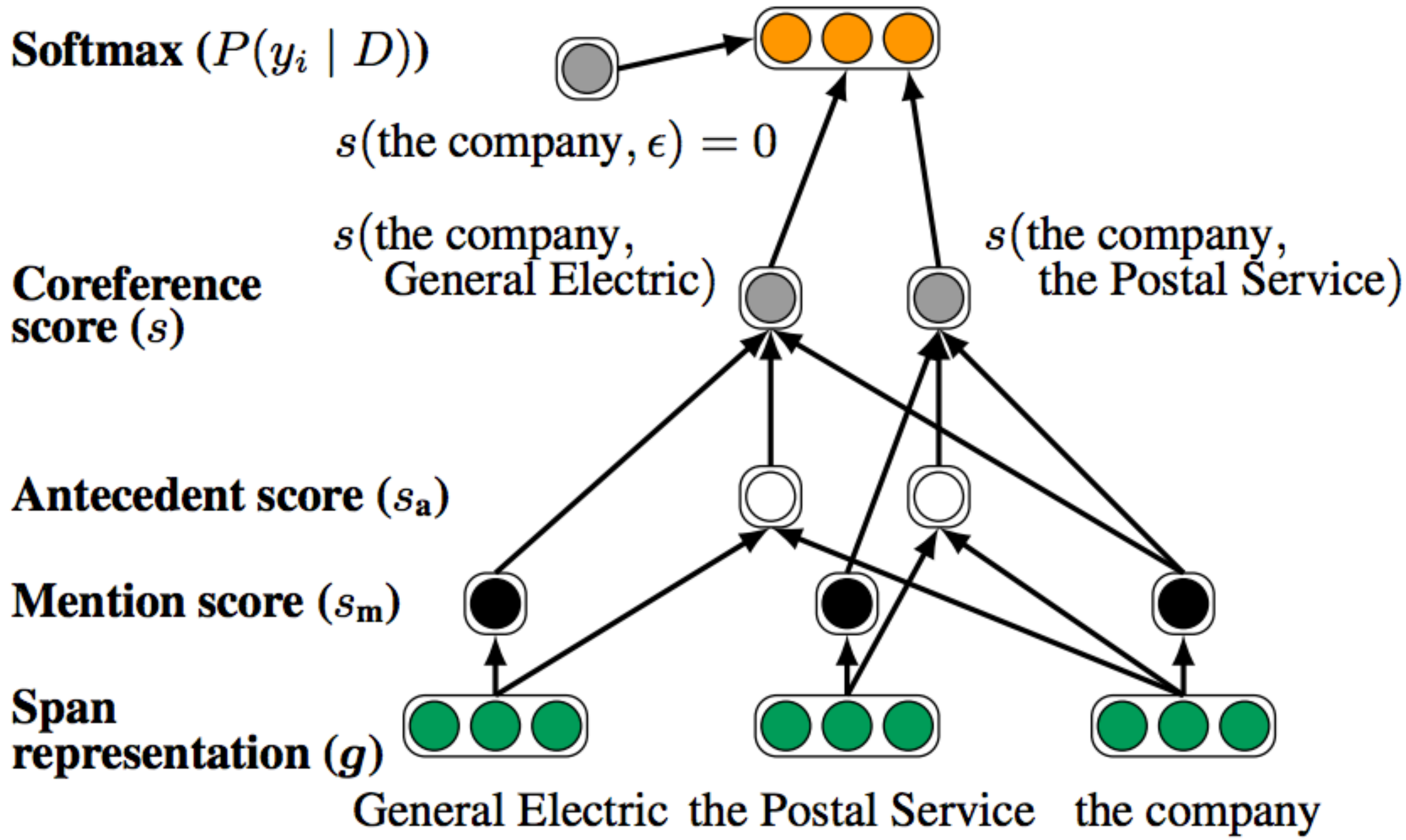
- 2 main contributions by this paper:
  - Can we represent all features with a more typical neural network embedding way?
  - Can neural network allow errors to flow end-to-end? All the way to mention detection?
    - This solves another type of error (span error), which is not previously handled.

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)

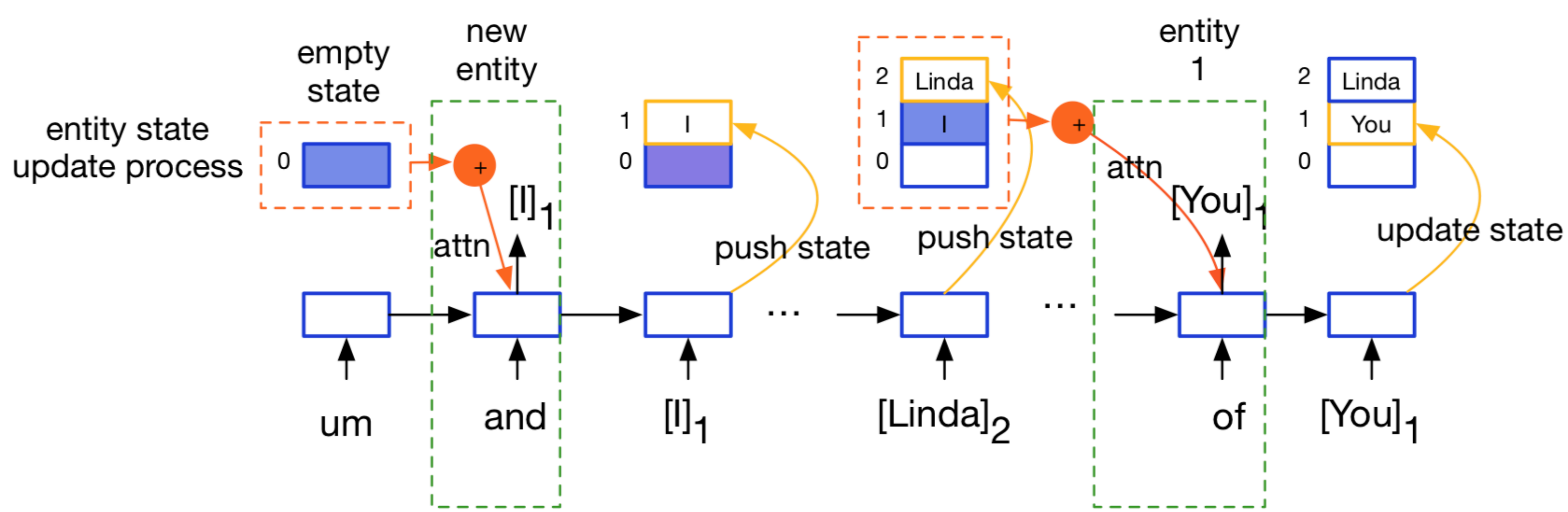


- Coreference model is similar to a mention ranking.
- Coreference score consist of multiple scores.
- Simple max-likelihood (not the cost sensitive method by Durrett, why?)

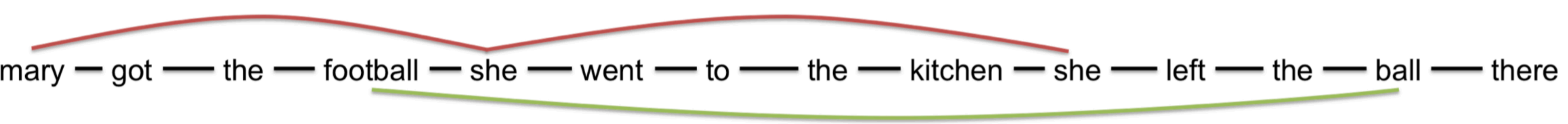
Using Coreference in Neural Models

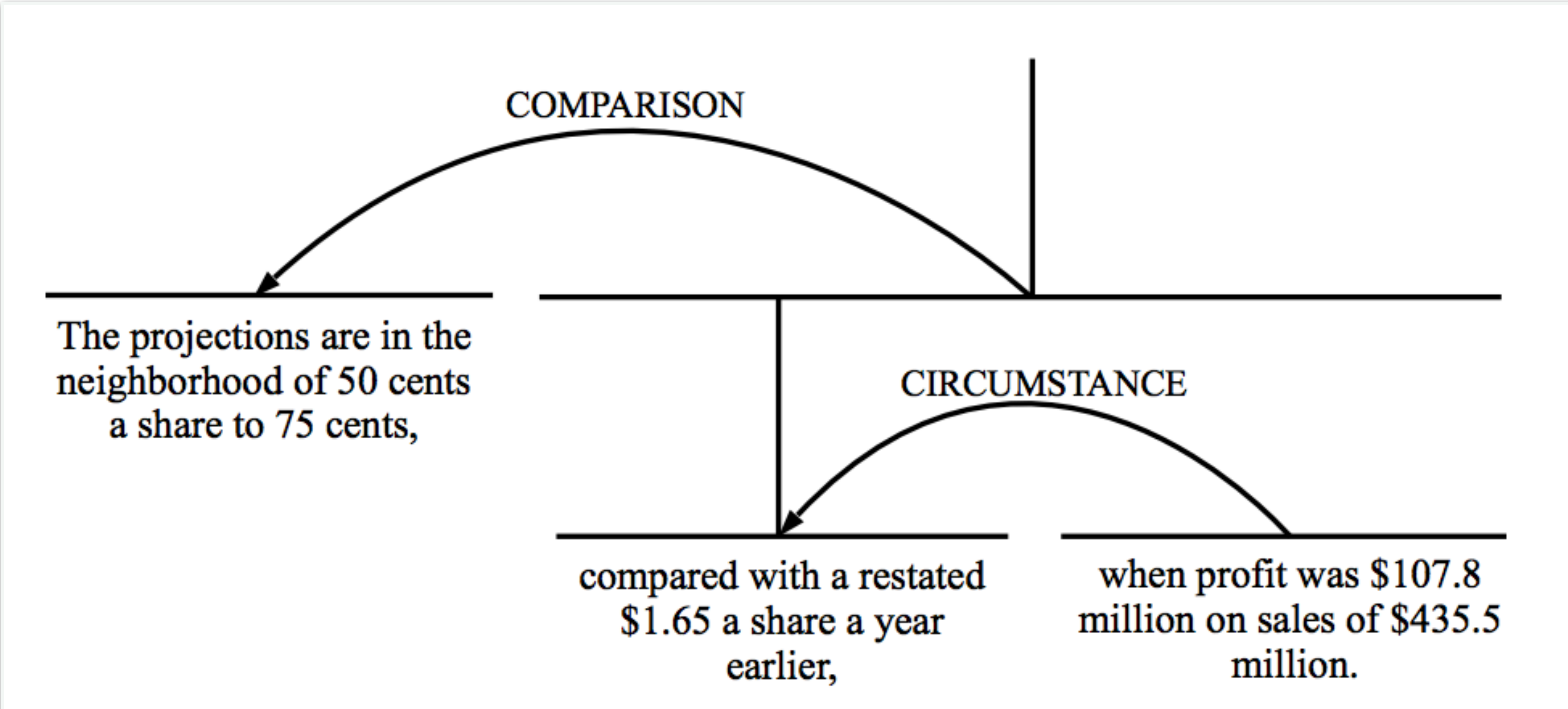
- Co-reference aware language modeling (Yang et al. 2017)

um and $[\text{I}]_1$ think that is whats - Go ahead $[\text{Linda}]_2$. Well and thanks goes to $[\text{you}]_1$ and to $[\text{the media}]_3$ to help $[\text{us}]_4$...So $[\text{our}]_4$ hat is off to all of $[\text{you}]_5$...



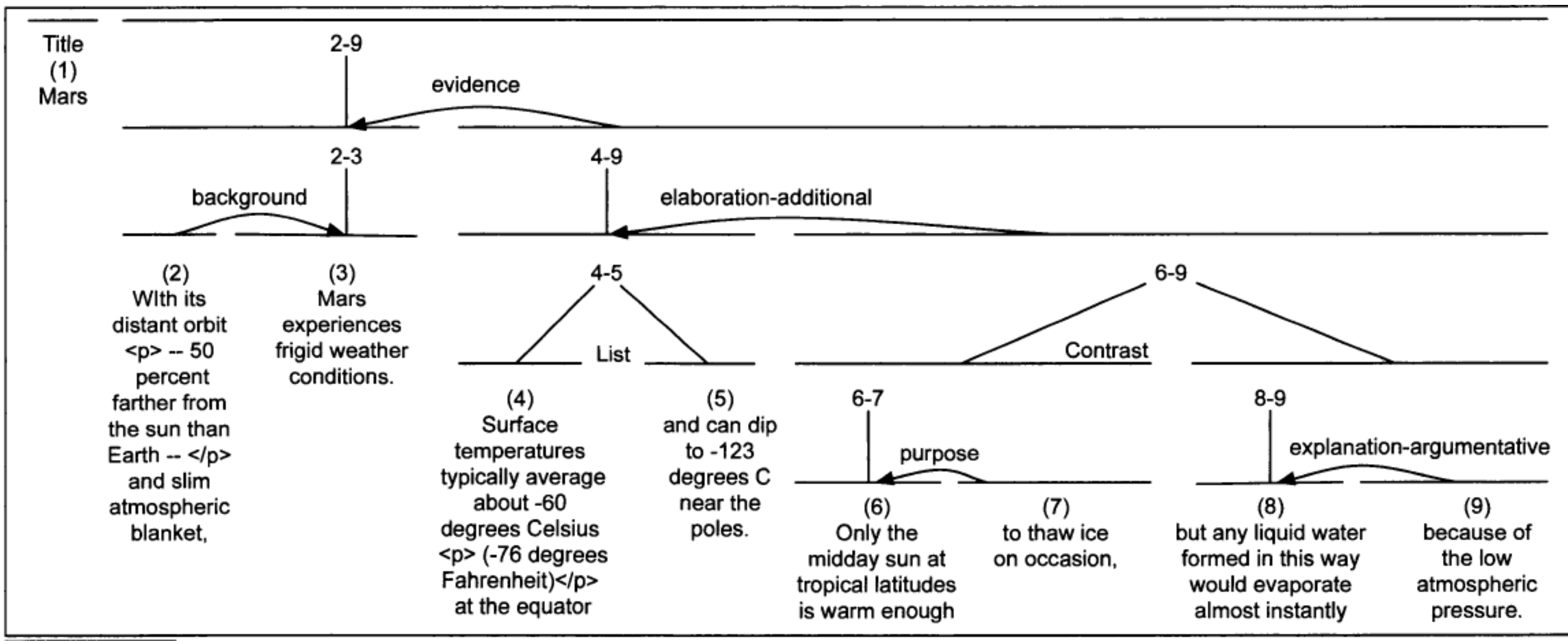
- Co-reference aware QA models (Dhingra et al. 2017)





Discourse Parsing

Document Problems: Discourse Parsing



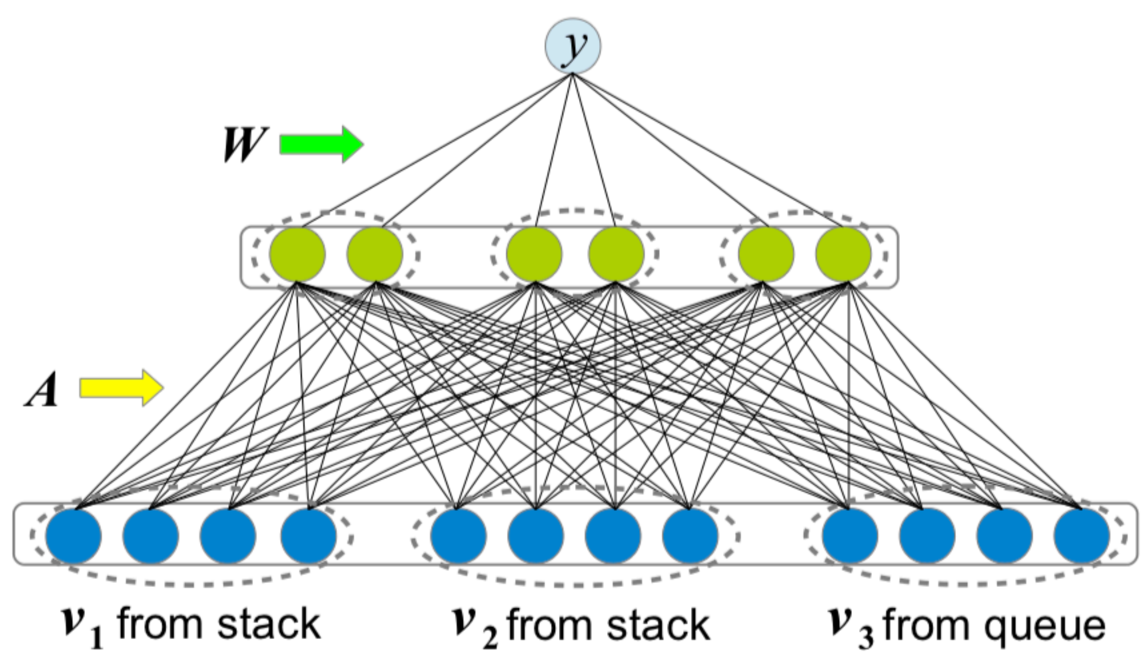
- Parse a piece of text into a relations between discourse units (EDUs).
- Researchers mainly used the Rhetorical Structure Theory (RST) formalism, which forms a tree of relations.

Example RST structures from Marcu (2000)

Shift-reduce Parsing Discourse Structure Parsing w/ Distributed Representations

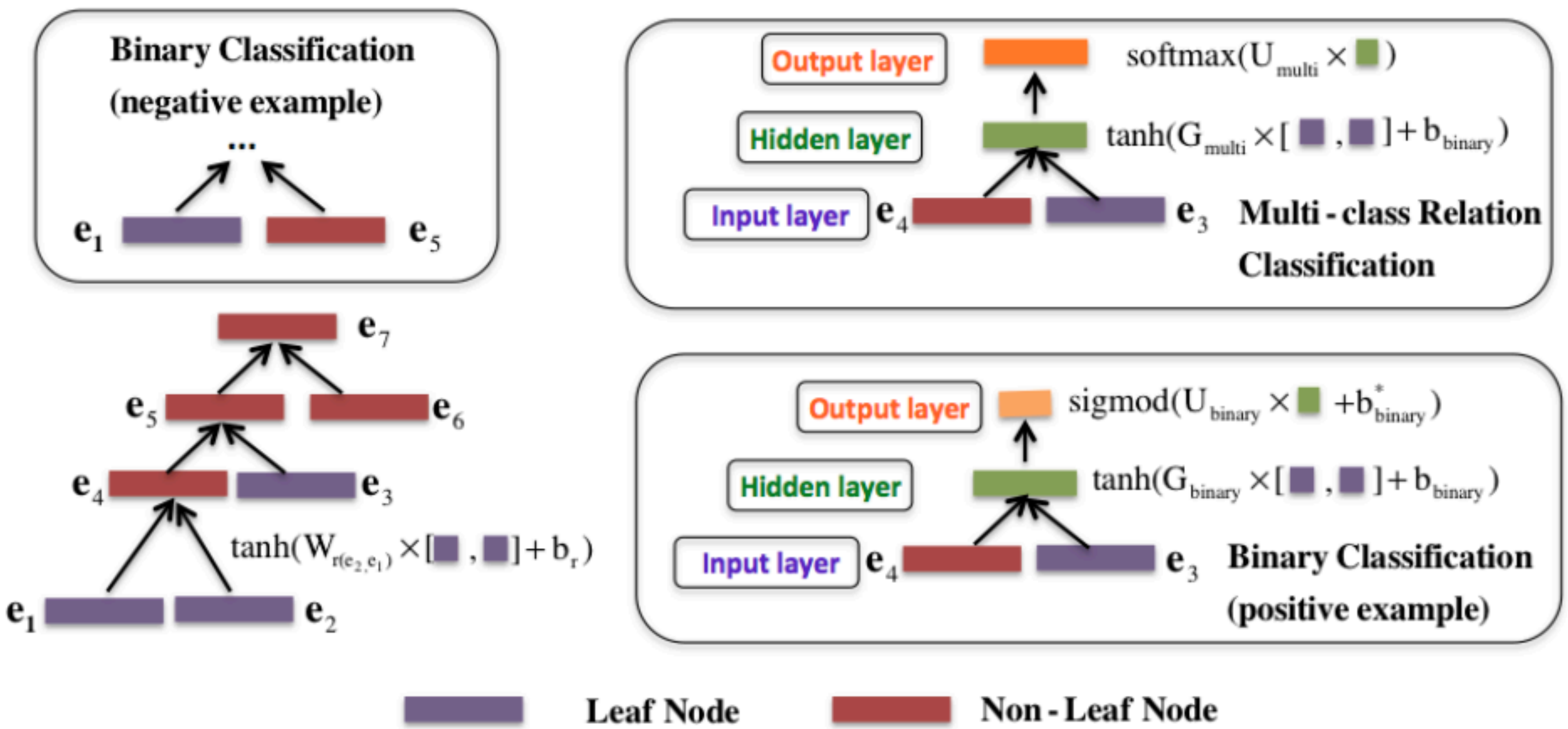
(Ji and Eisenstein 2014)

- Shift-reduce parser with features from 2 stack elements and queue element
- Project features into distributed space for better accuracy



Recursive Deep Models for Discourse Parsing

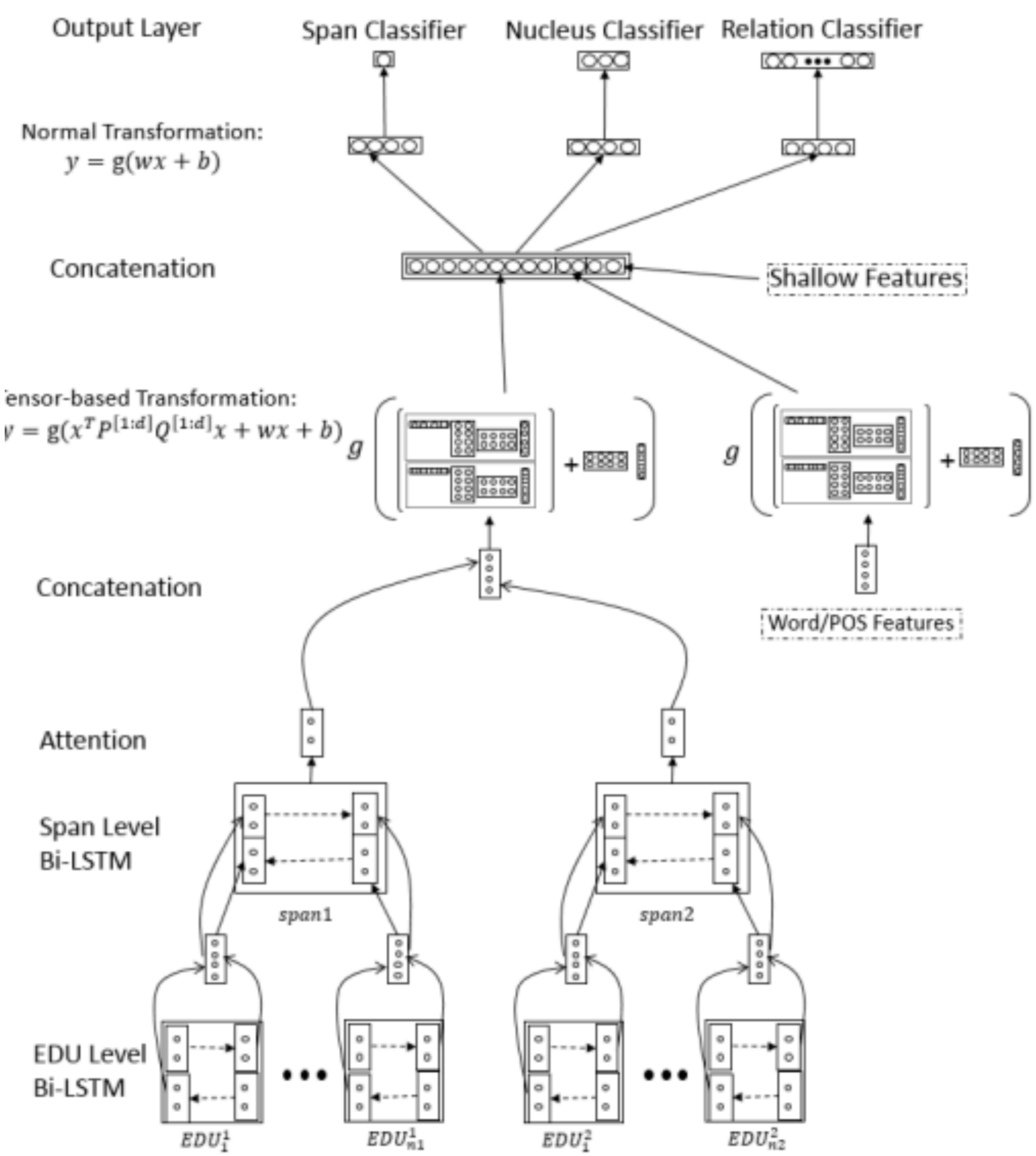
Li et.al (2014)



- 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

Discourse Parsing w/ Attention-based Hierarchical Neural Networks

Li et.al (2016)



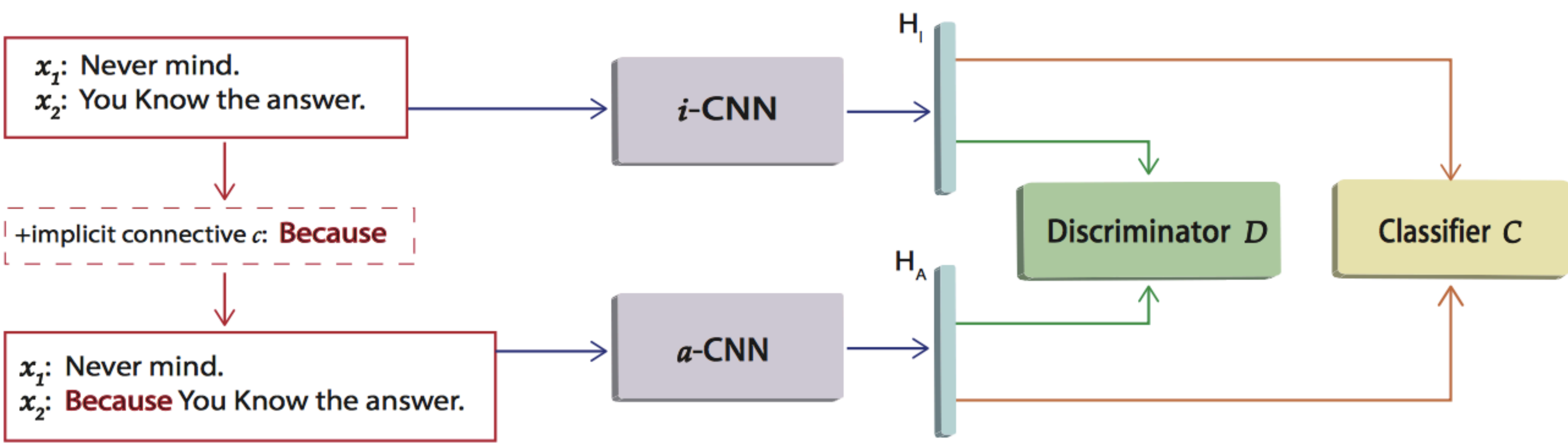
- Hierarchical bi-LSTM to learn composition scoring.
- Augmented with attention mechanism. (Span is long)
- 2 Bi-LSTMs: first used to capture the representation of a EDU, then combine EDU representation into larger representation
- CKY Parsing

Implicit Discourse Connection Classification w/ Adversarial Objective

(Qin et al. 2017)

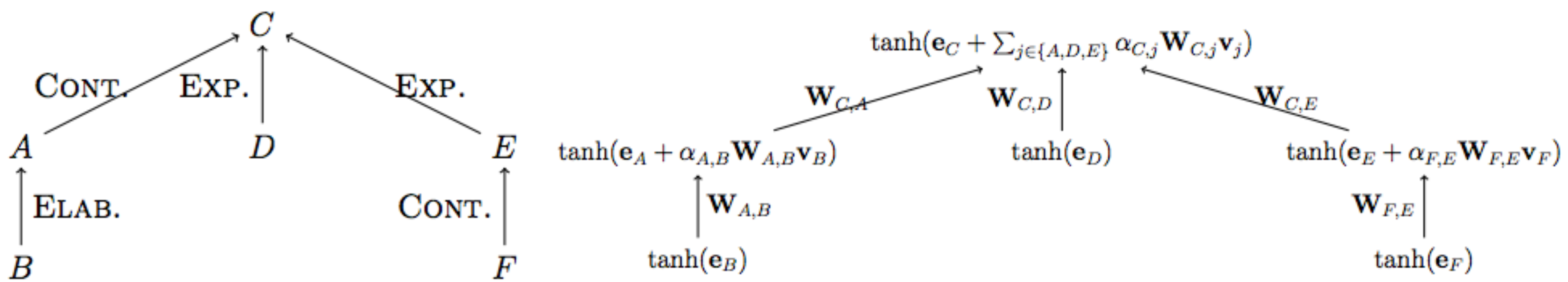
- Idea: implicit discourse relations are not explicitly marked, but would like to detect them if they are

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



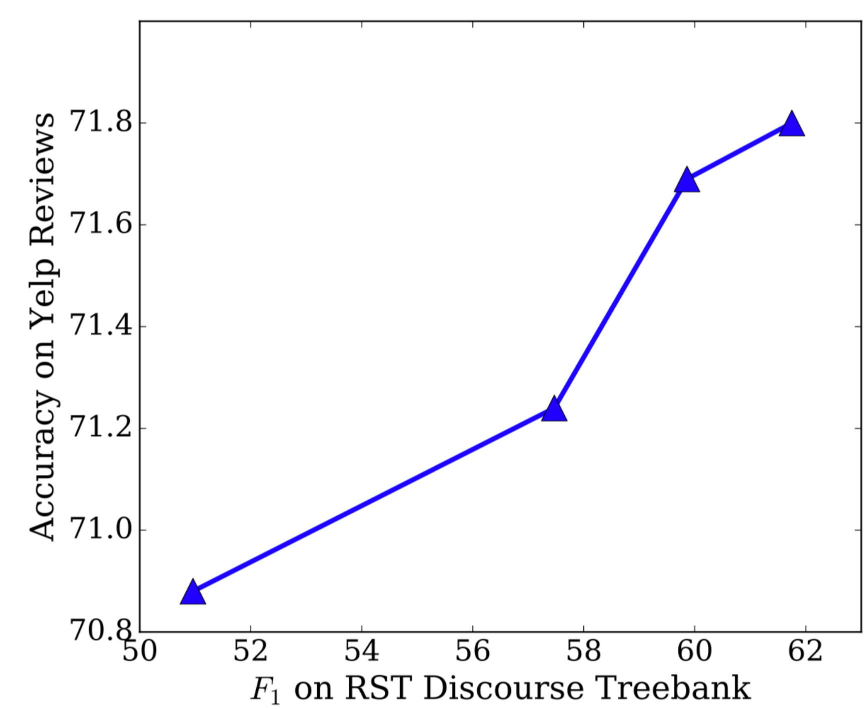
Uses of Discourse Structure in Neural Models

- Discourse-structured classification with neural models (Ji and Smith 2017)



(a) dependency structure (b) recursive neural network structure

- Good results, and more interestingly, discourse parsing accuracy very important!



Questions?