CS11-747 Neural Networks for NLP

Document Level Models

Graham Neubig

Carnegie Mellon University
Language Technologies Institute

Site
https://phontron.com/class/nn4nlp2021/

(w/ thanks for many Slides from Zhengzhong Liu)

Some NLP Tasks we've Handled

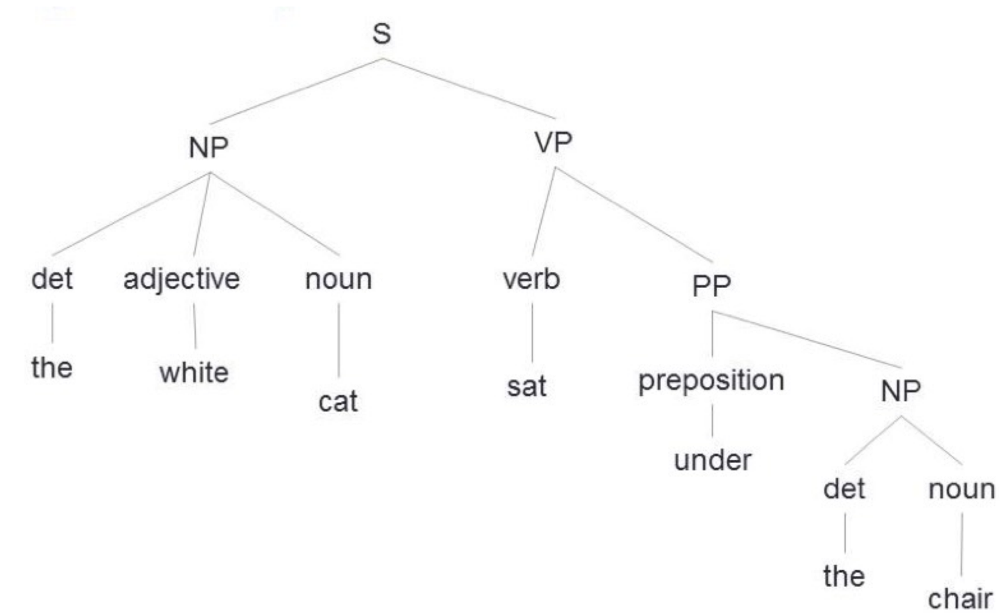
Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, 'and what is the use of a book,' thought Alice 'without pictures or conversation?'

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

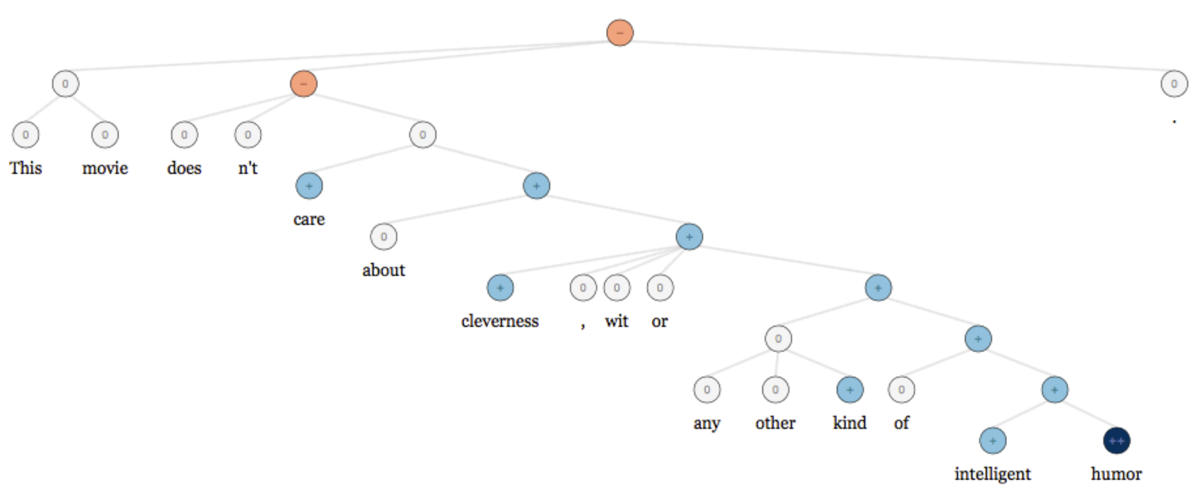
$P(w_{i+1} = \text{bank} \mid w_i = \text{the}) = 1/3$
$P(w_{i+1} = \text{book} \mid w_i = \text{the}) = 1/3$
$P(w_{i+1} = \text{use} \mid w_i = \text{the}) = 1/3$

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

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 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?'

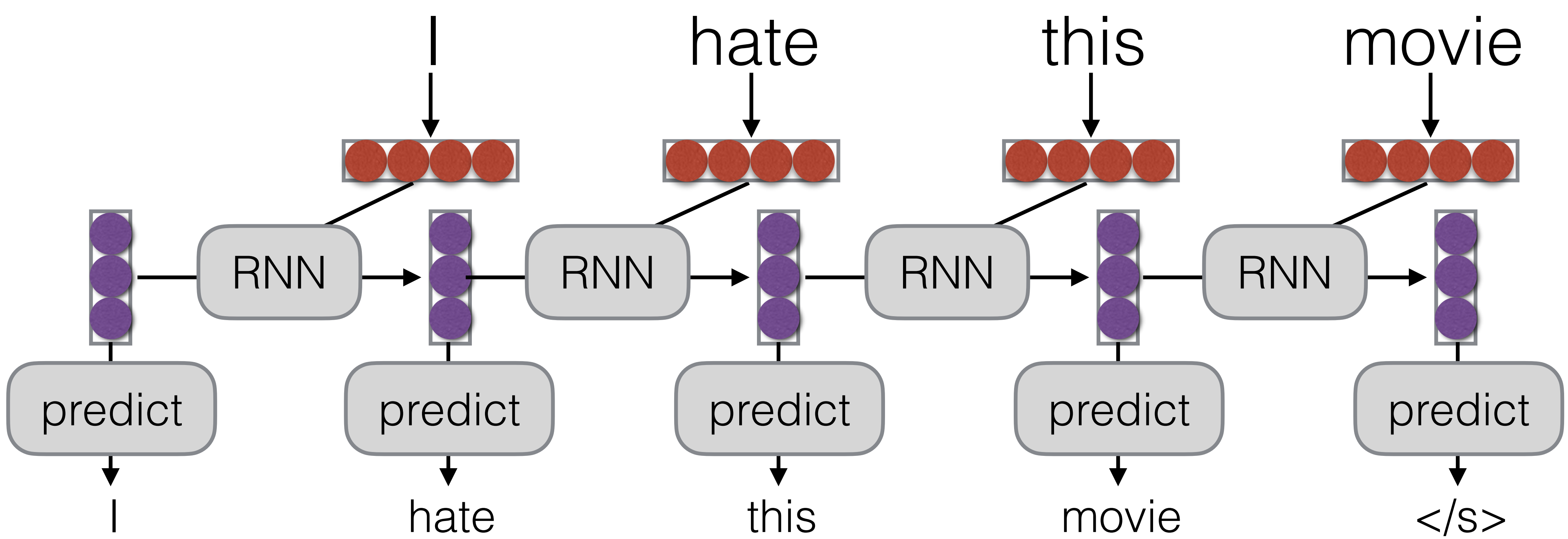
Document Level Language Modeling

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.

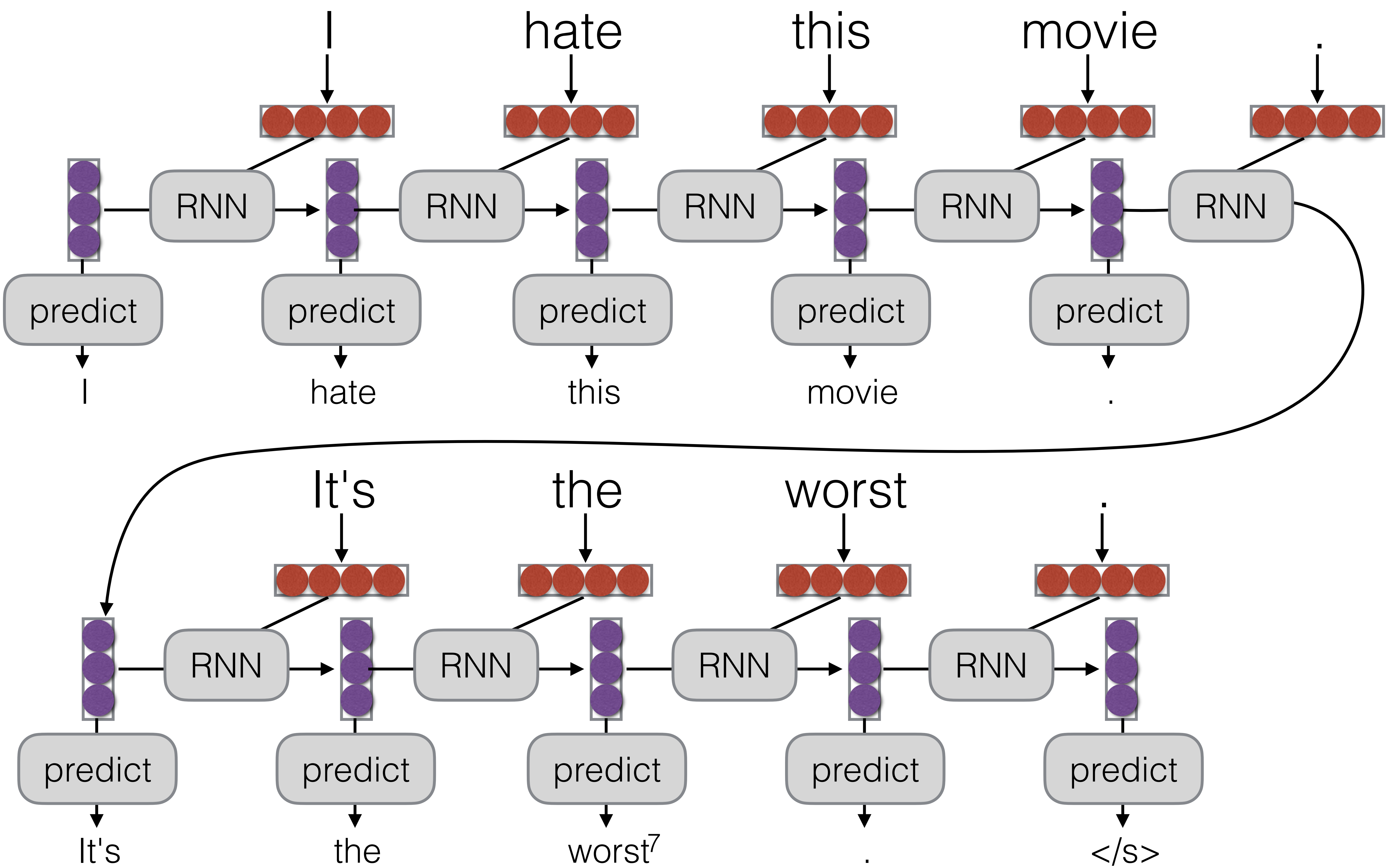
Remember: Modeling using Recurrent Networks

- Model passing previous information in hidden state



Simple: Infinitely Pass State

(Mikolov et al. 2011)

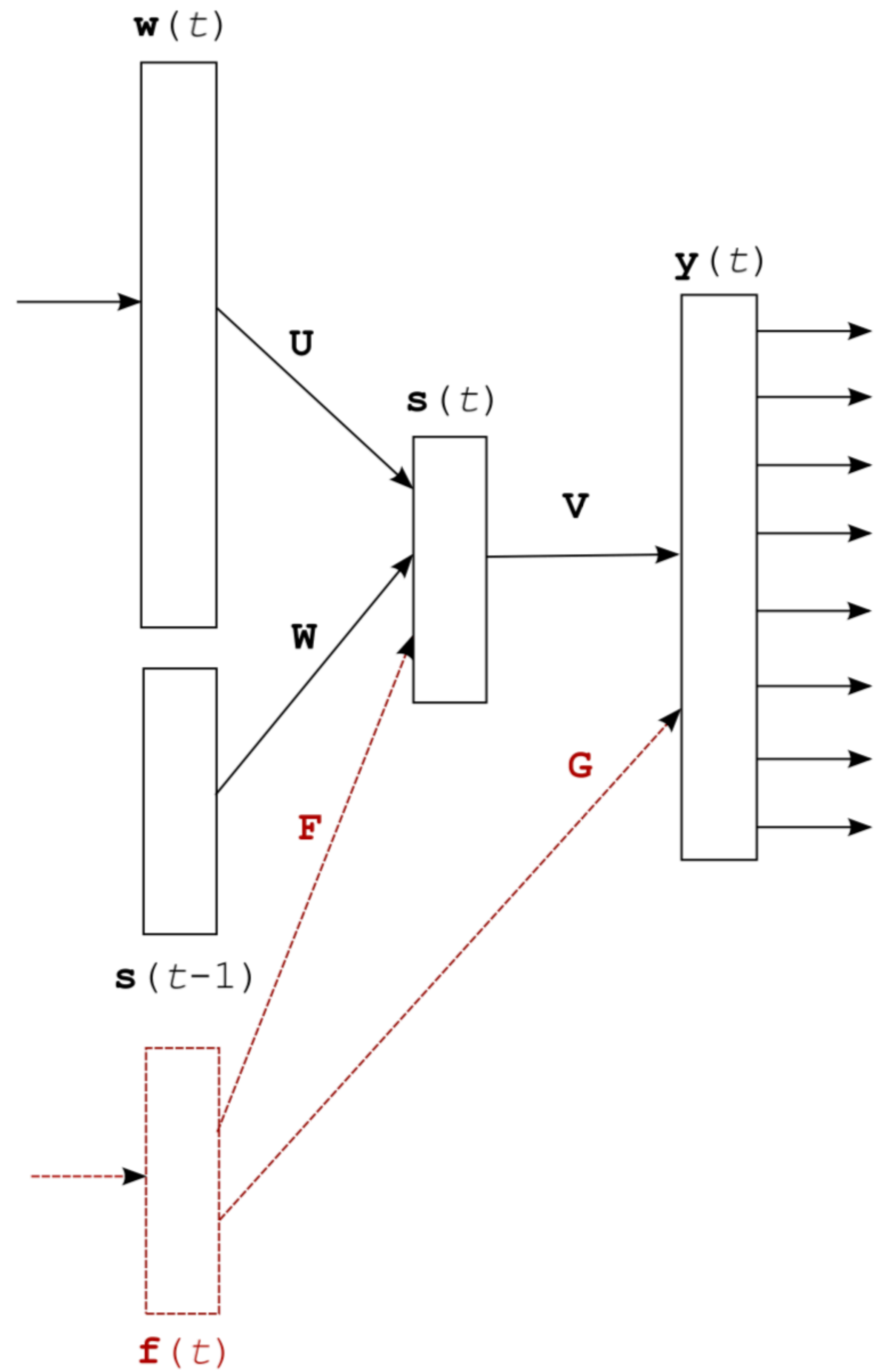


Separate Encoding for Coarse-grained Document Context

(Mikolov & Zweig 2012)

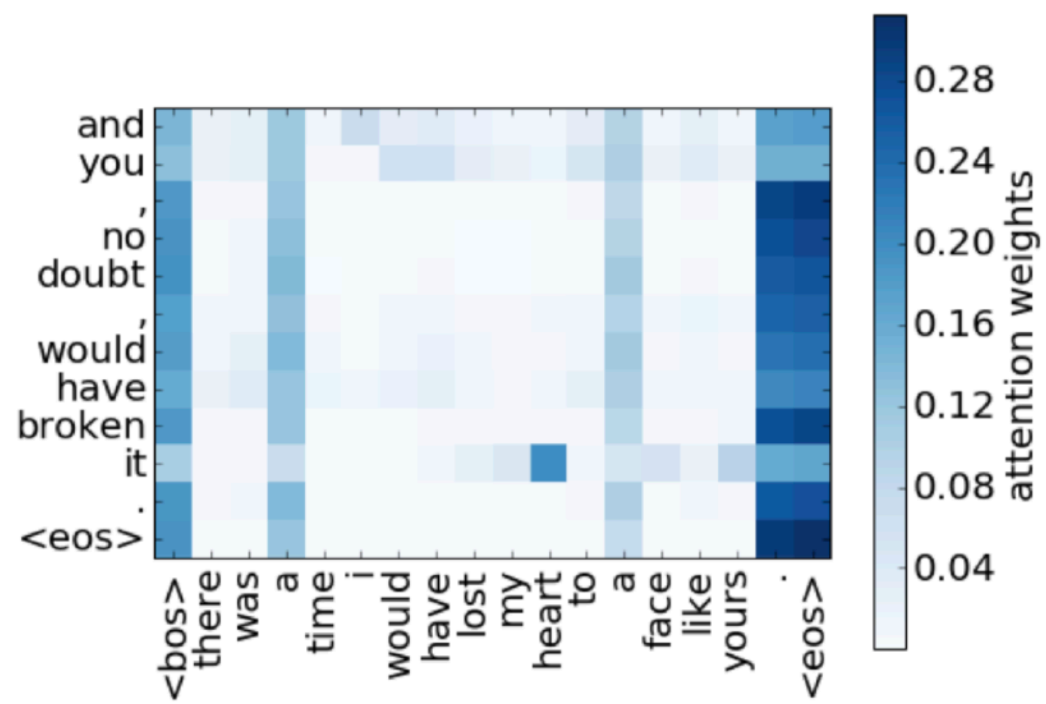
- One big RNN for local and global context tends to miss out on global context (as local context is more predictive)

- Other attempts try to incorporate document-level context explicitly



Self-attention/Transformers Across Sentences

- Simply self-attend to all previous words in the document (e.g. Voita et al. 2018)
- + Can relatively simply use document-level context
- + Can learn interesting phenomena (e.g. co-reference)

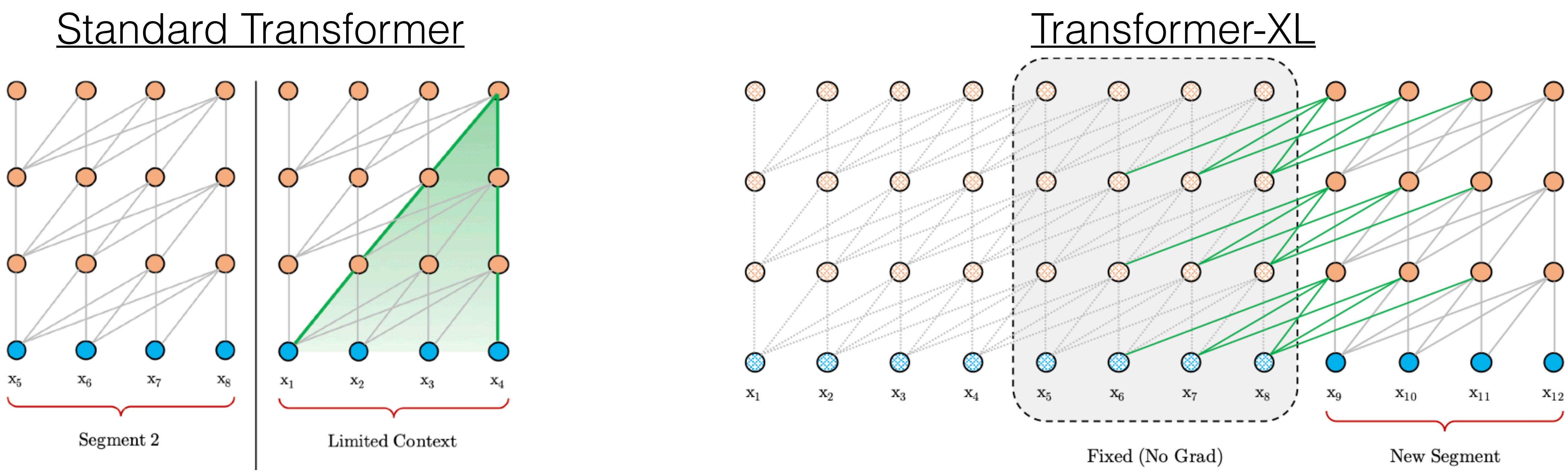


- - Computation is quadratic in sequence length!

Transformer-XL: Truncated BPTT+Transformer

(Dai et al. 2019)

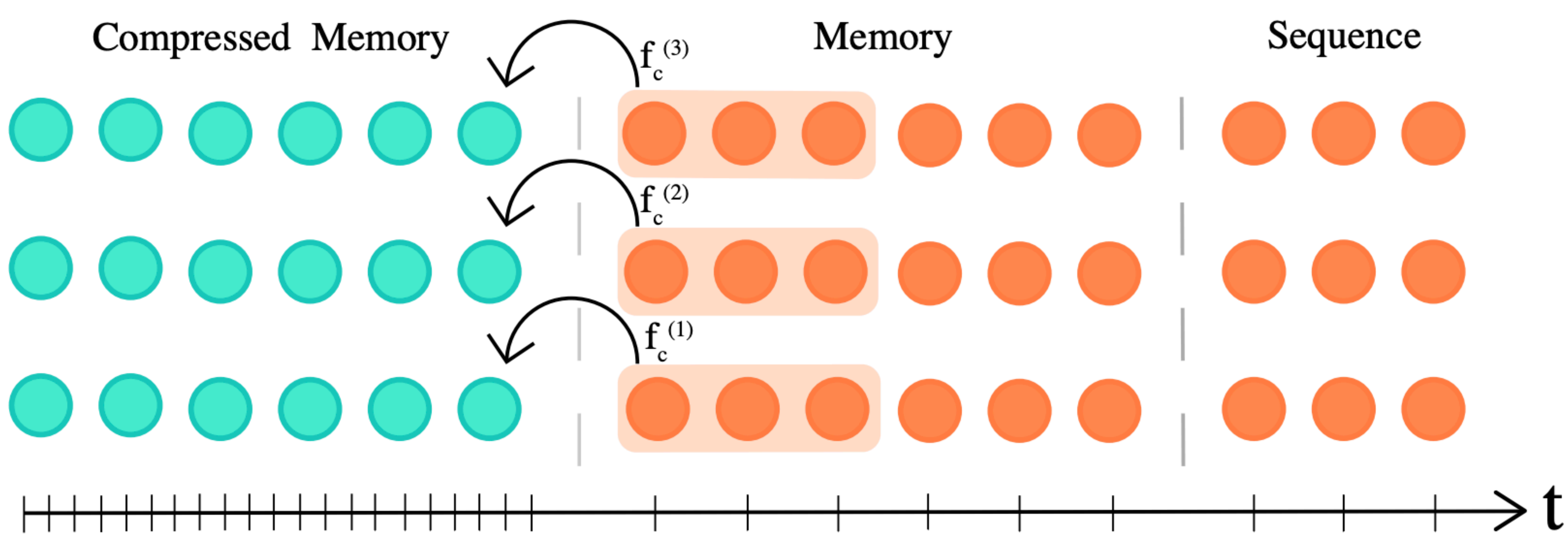
- Idea: attend to fixed **vectors** from the previous sentence (Dai et al. 2019)



- Like truncated backprop through time for RNNs; can use previous states, but not backprop into them

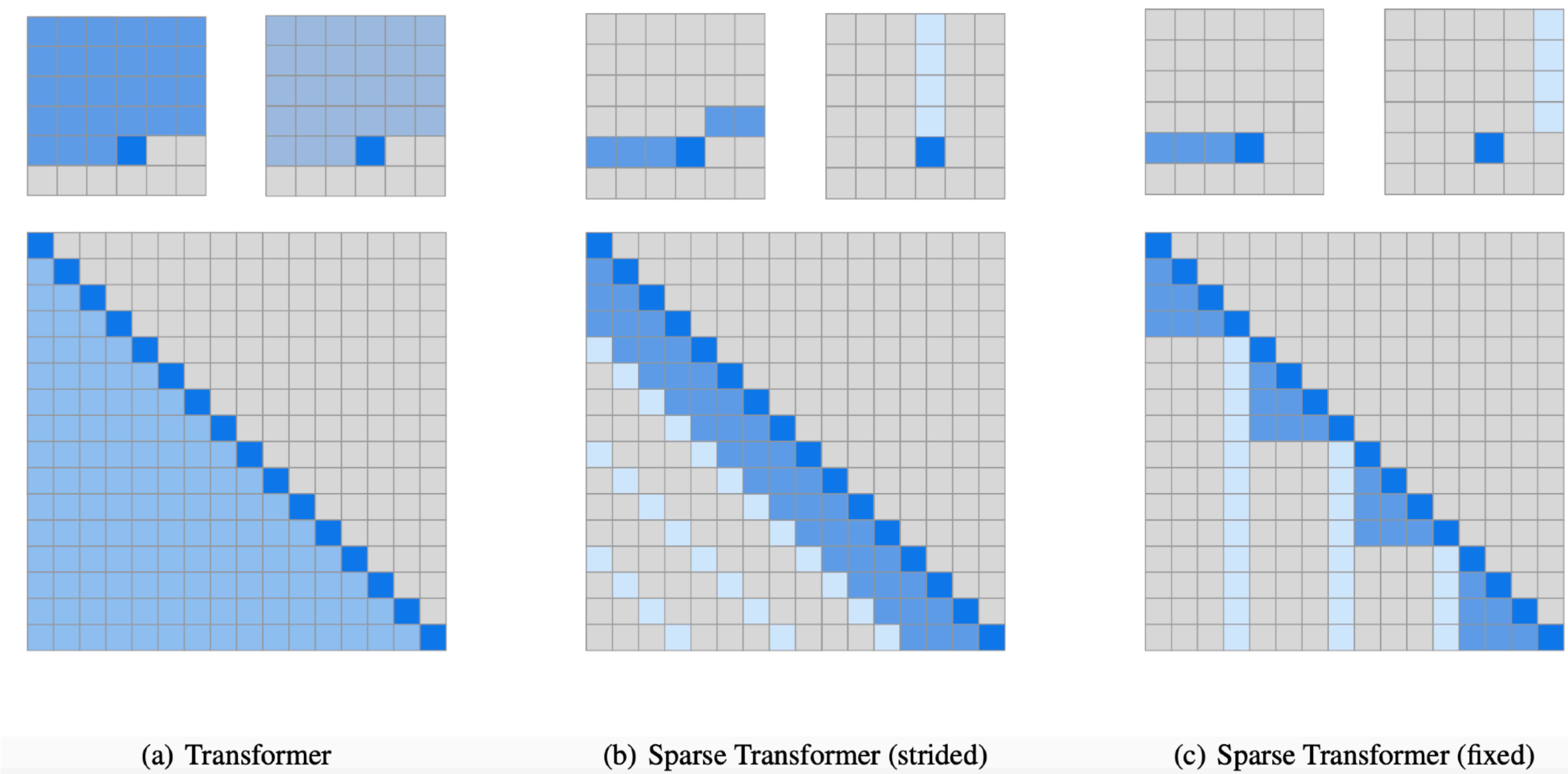
Compressing Previous States

- Add a "strided" compression step over previous states (Lillicrap et al. 2019)



Sparse Transformers
(Child et al. 2019)

- Add "stride", only attending to every *n* previous states



Adaptive Span Transformers

- Can make the span adaptive attention head by attention head some are short, some long (Sukhbaatar et al. 2019)

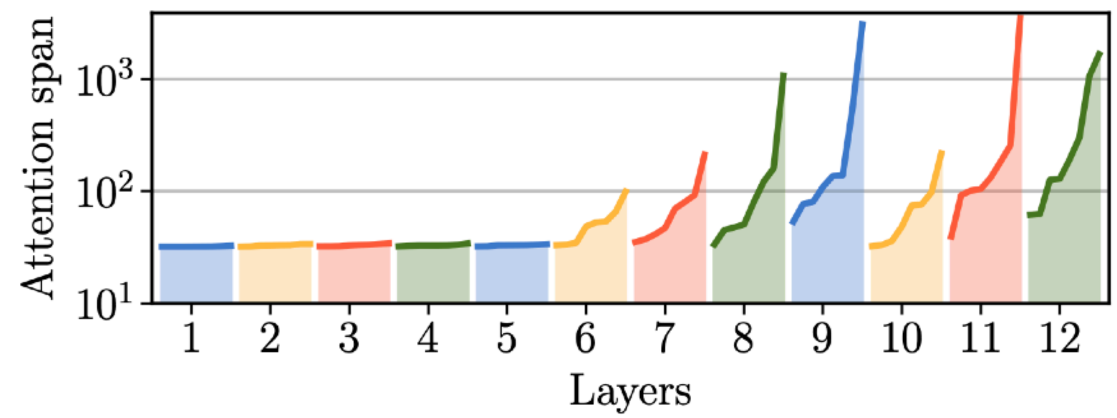
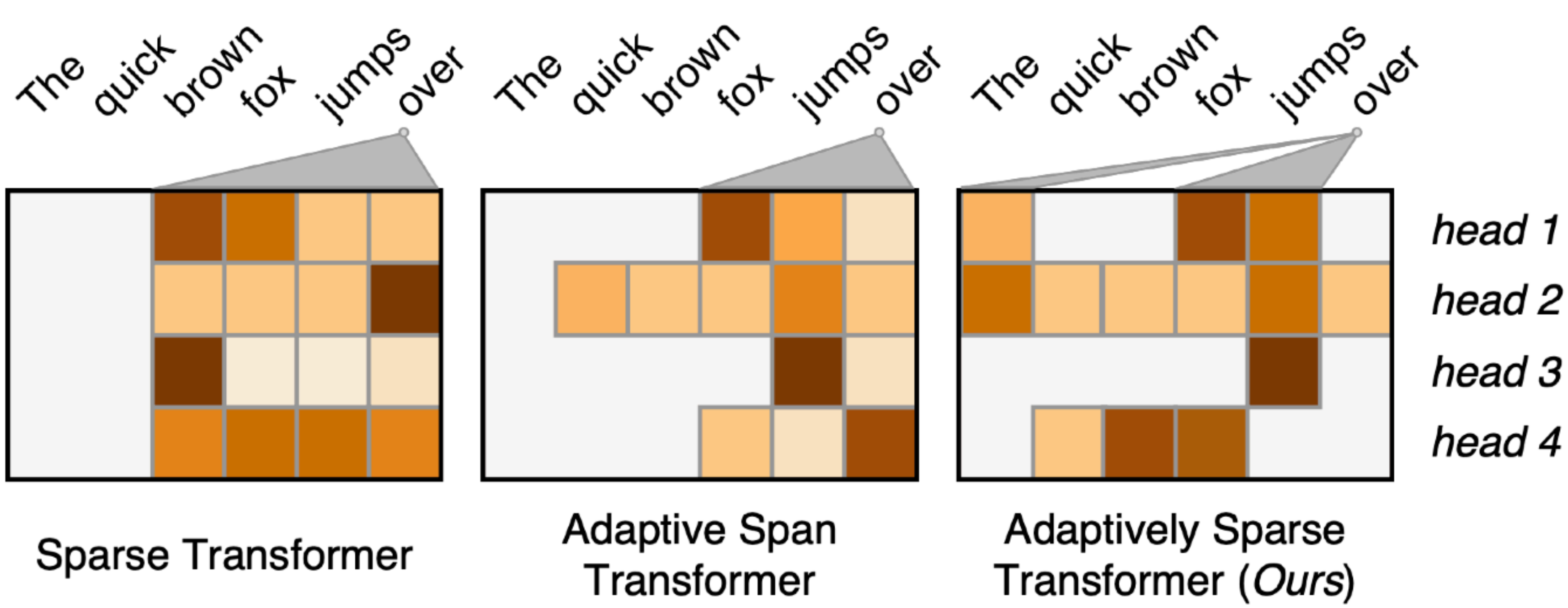


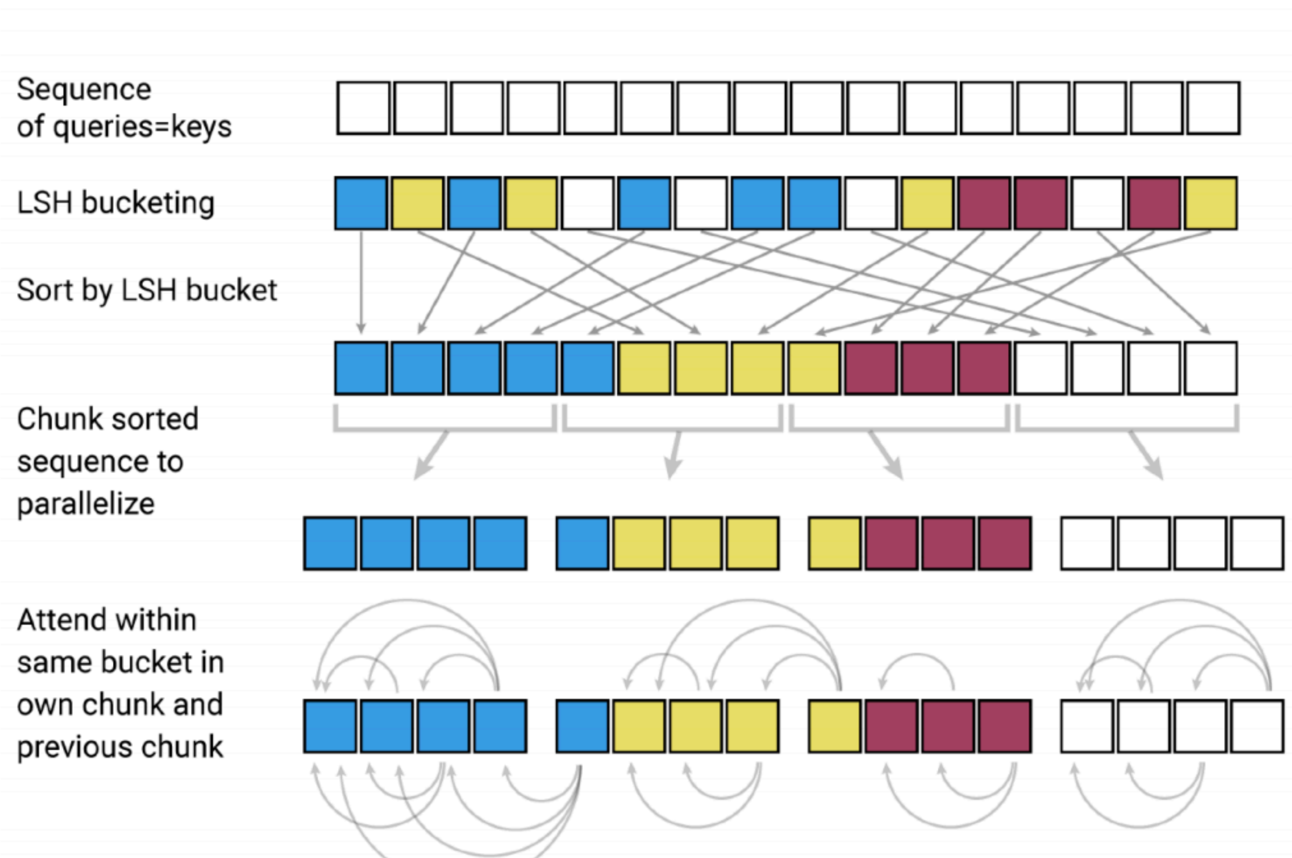
Figure 4: Adaptive spans (in log-scale) of every attention heads in a 12-layer model with span limit $S = 4096$. Few attention heads require long attention spans.

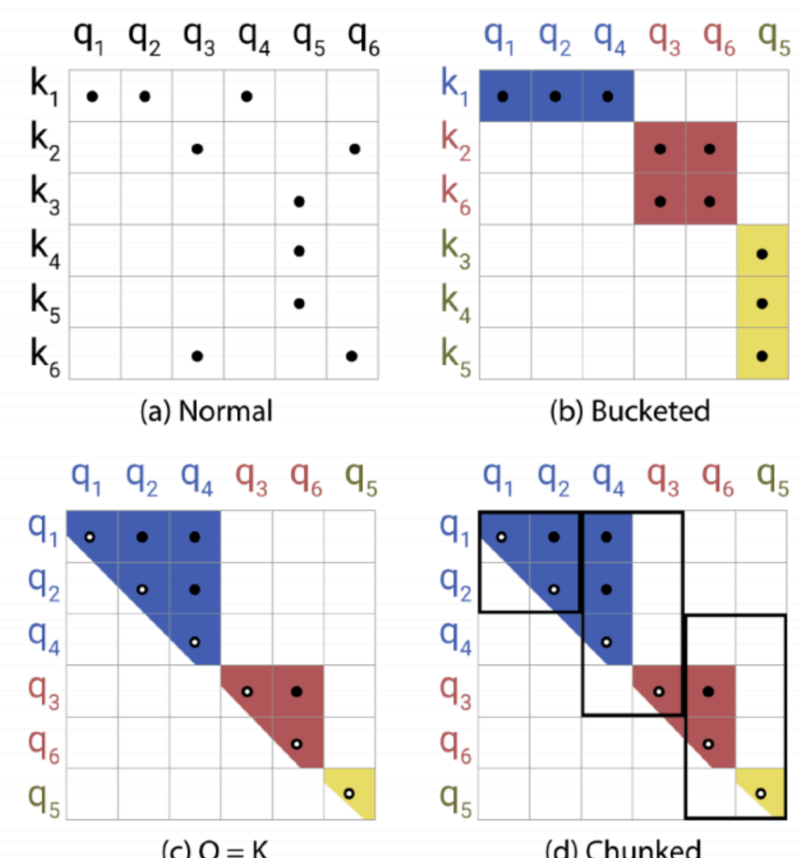
- Can be further combined with sparse computation (Correira et al. 2019)



Reformer: Efficient Adaptively Sparse Attention

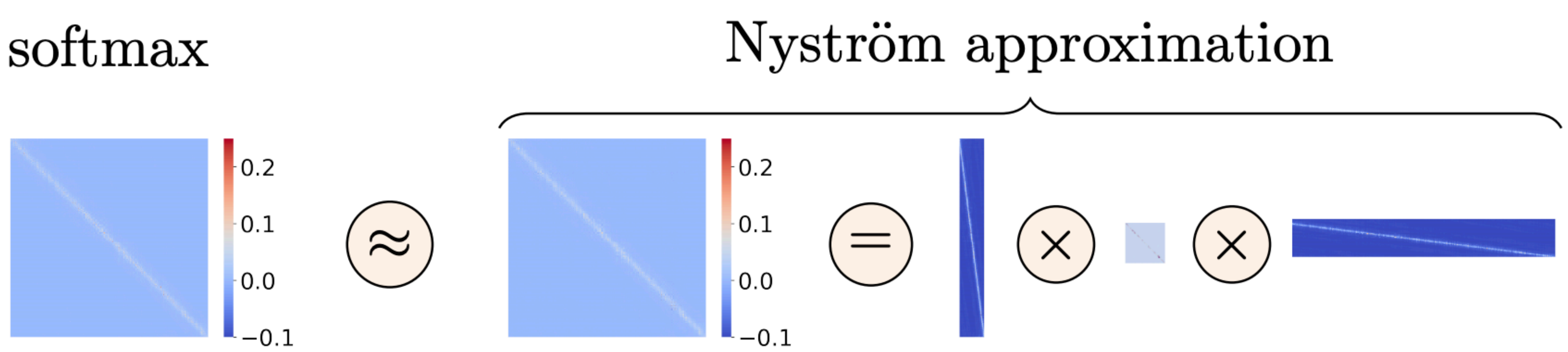
- Chicken-and-egg problem in sparse attention:
  - Can sparsify relatively low-scoring values to improve efficiency
  - Need to calculate all values to know which ones are relatively low-scoring
- **Reformer** (Kitaev et al. 2020): efficient calculation of sparse attention through
  - Shared key and query parameters to put key and query in the same space
  - Locality sensitive hashing to efficiently calculate high-scoring attention weights
  - Chunking to make sparse computation more GPU friendly





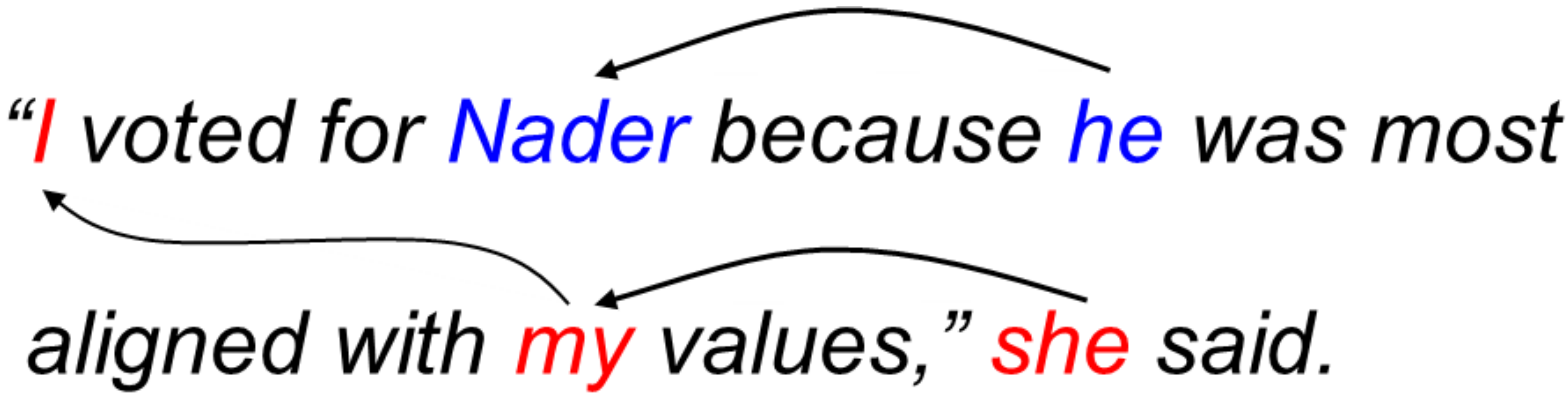
Low-rank Approximation

- Calculating the attention matrix is expensive, can it be predicted with a low-rank matrix?
- **Linformer:** Add low-rank linear projections into model (Wang et al. 2020)
- **Nystromformer:** Approximate using the Nystrom method, sampling "landmark" points (Xiong et al. 2021)



How to Evaluate Document-level Models?

- Simple: Perplexity, classification over long documents
- More focused:
  - Sentence scrambling (Barzilay and Lapata 2008)
  - Final sentence prediction (Mostafazadeh et al. 2016)
  - Final word prediction (Paperno et al. 2016)
- Composite benchmark containing several task: Long range arena (Tay et al. 2020)



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*...

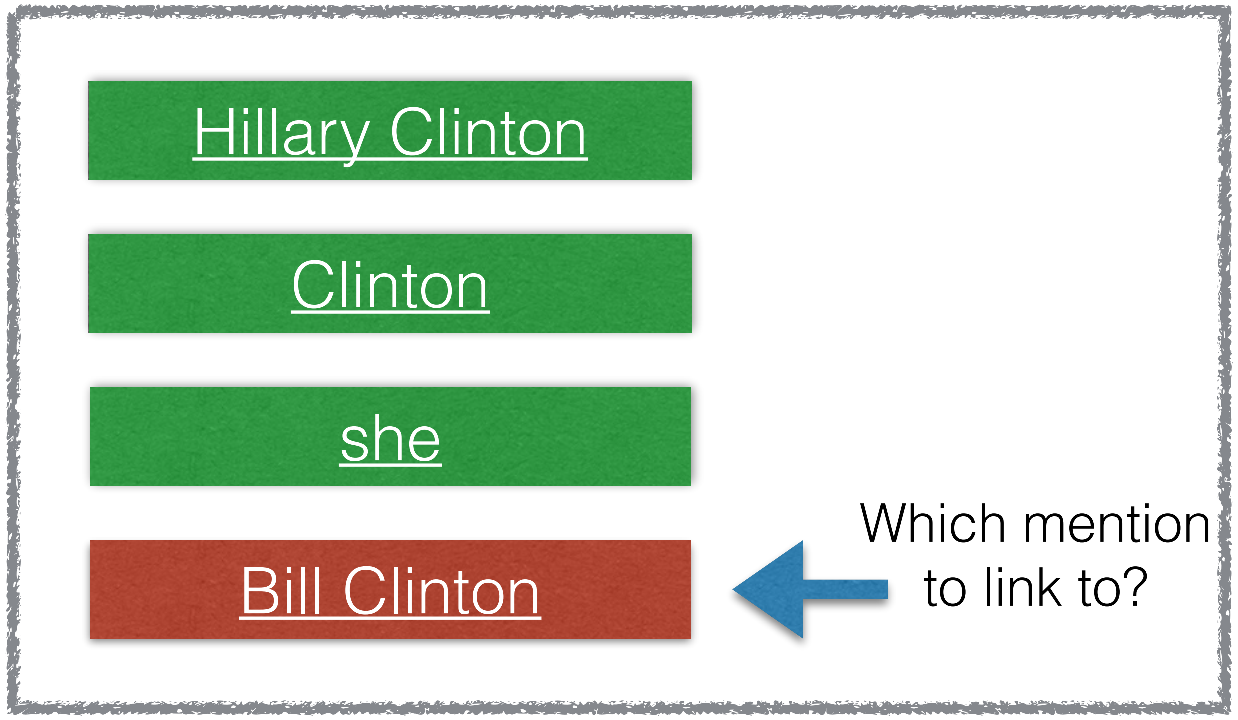
- One may think coreference is simply a clustering problem of given Noun Phrases.
  - Detecting relevant noun phrases is a difficult and important step.
  - Knowing the correct noun phrases affect the result a lot.
  - Normally done as a preprocessing step.

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

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.

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*...

✔: 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.

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

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

Advantages of Neural Network Models for Coreference

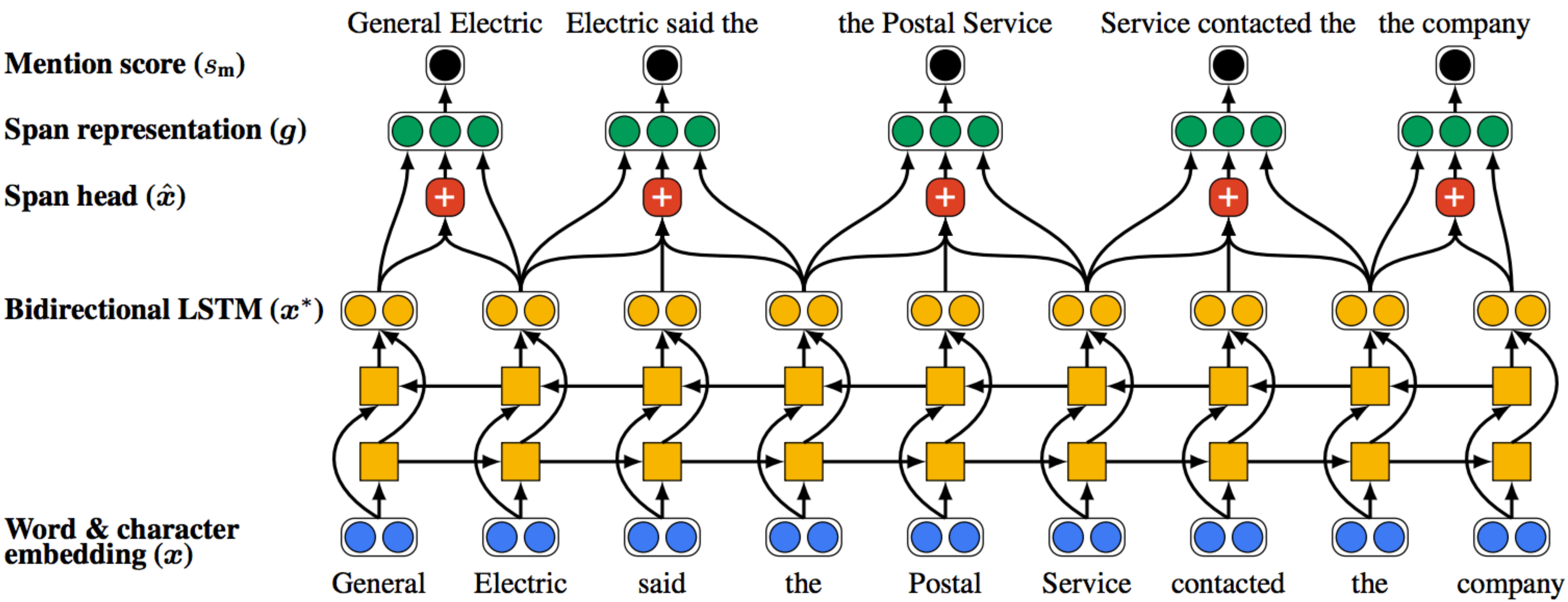
- **Learn the features** with embeddings since most of them can be captured by surface features.
- **Train towards the metric** using reinforcement learning or margin-based methods.
- **Jointly perform mention detection** and clustering.

End-to-End Neural Coreference

Lee et.al (2017)

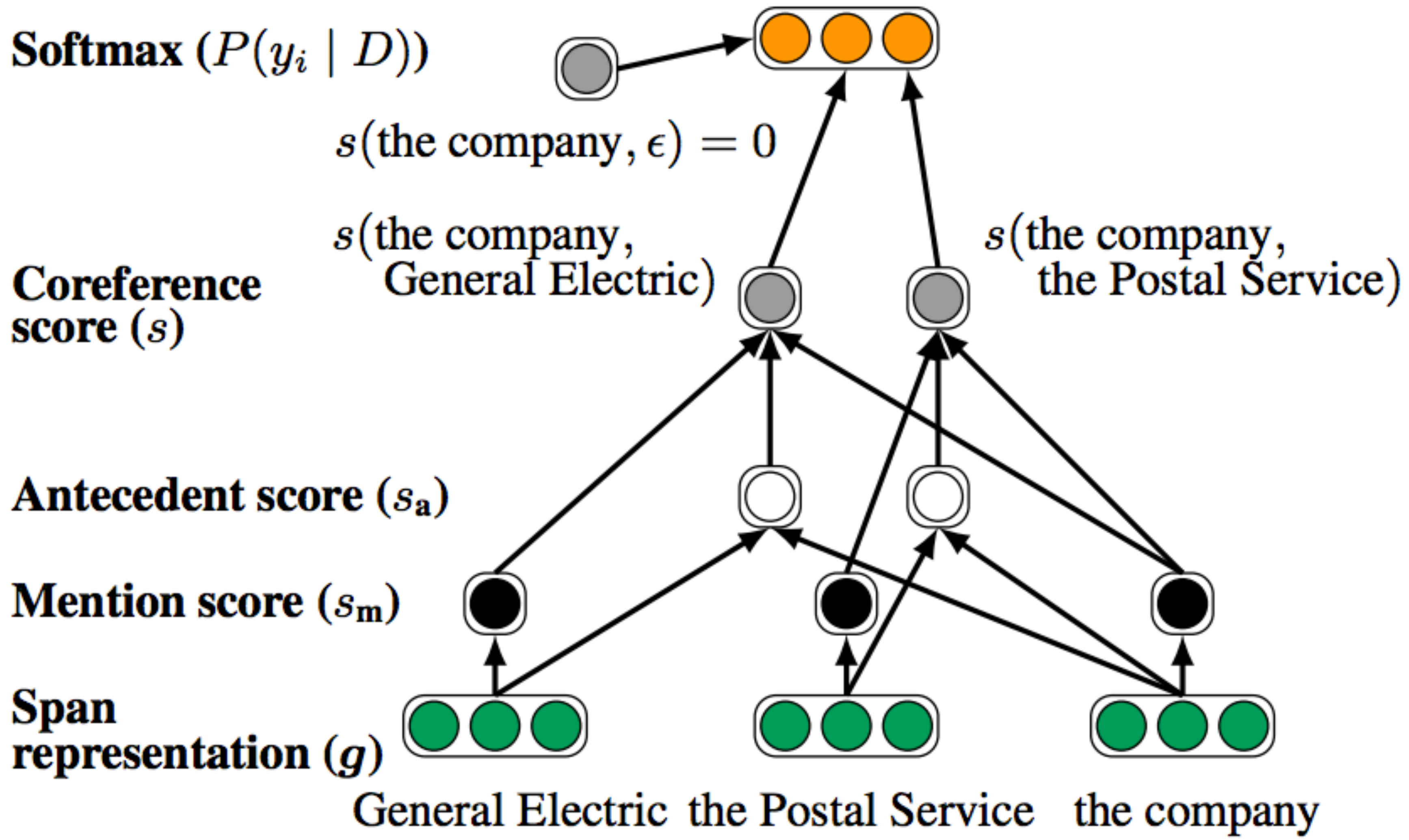
- 2 main contributions by this paper:
  - Can we represent all features with a more typical neural network embedding way?
  - Can neural network allow errors to flow end-to-end? All the way to mention detection?
    - This solves another type of error (span error), which is not previously handled.

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)

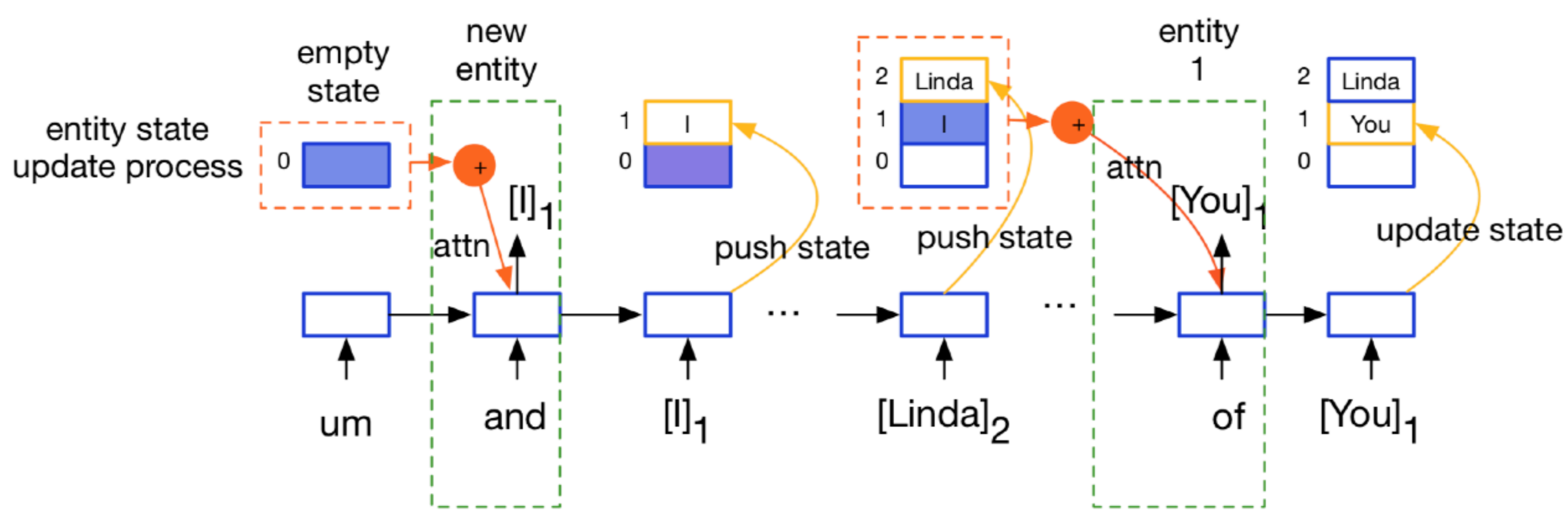


- Coreference model is similar to a mention ranking.
- Coreference score consist of multiple scores.
- Simple max-likelihood

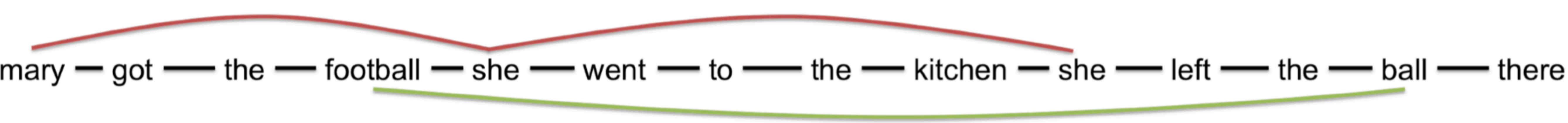
Using Coreference in Neural Models

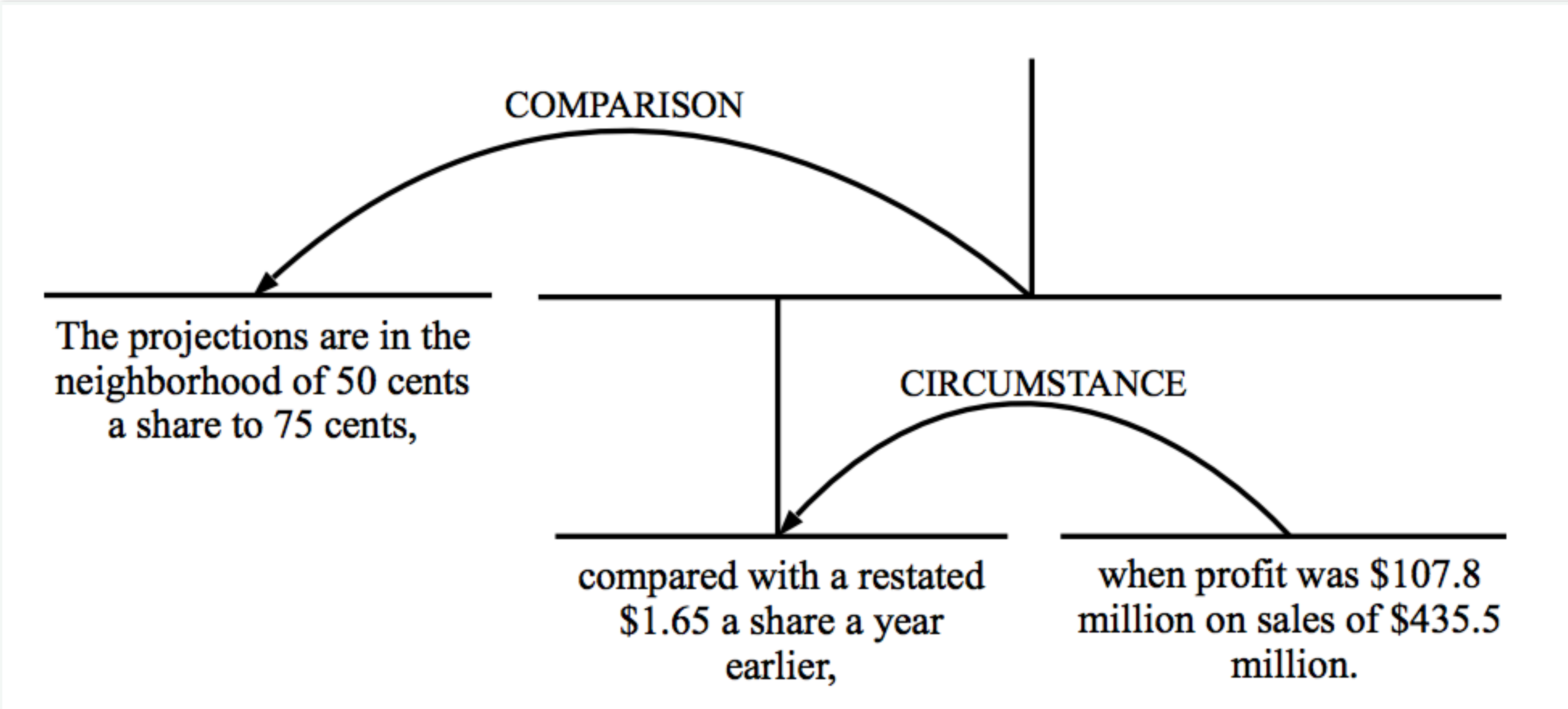
- Co-reference aware language modeling (Yang et al. 2017)

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



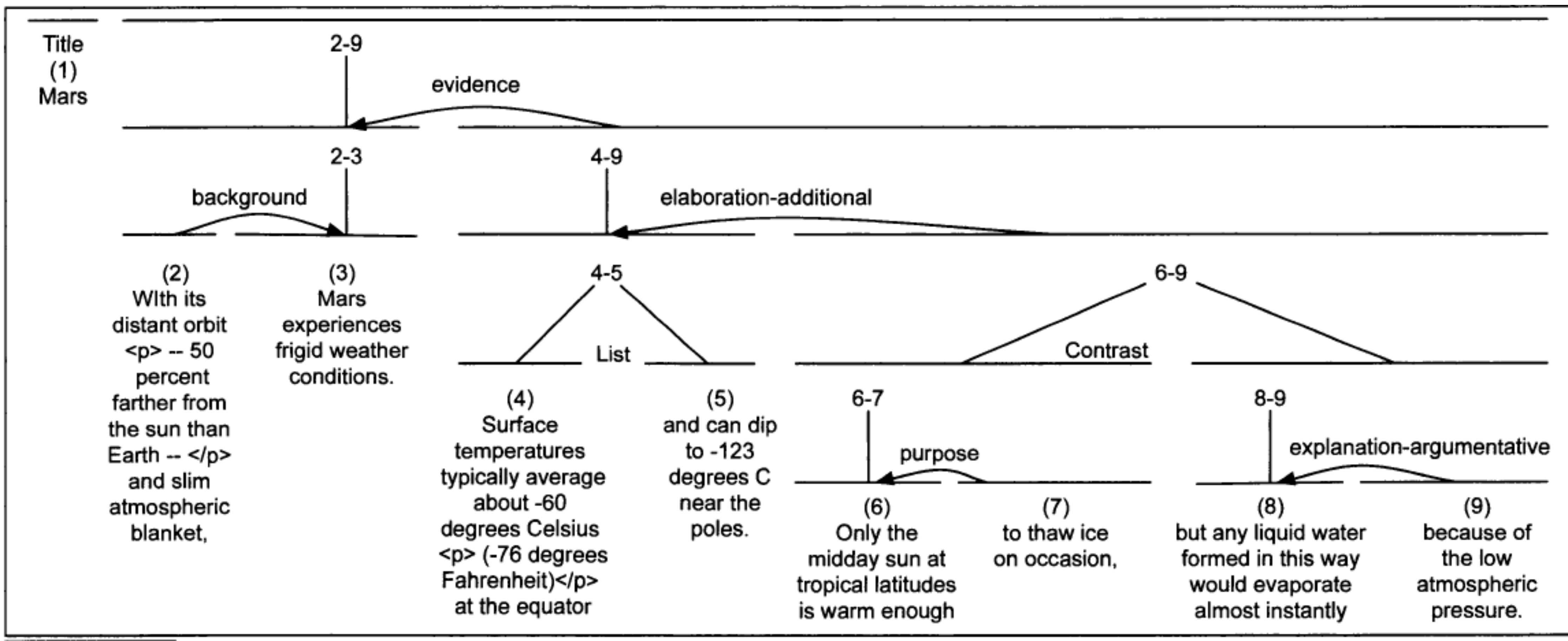
- Co-reference aware QA models (Dhingra et al. 2017)





Discourse Parsing

Document Problems: Discourse Parsing

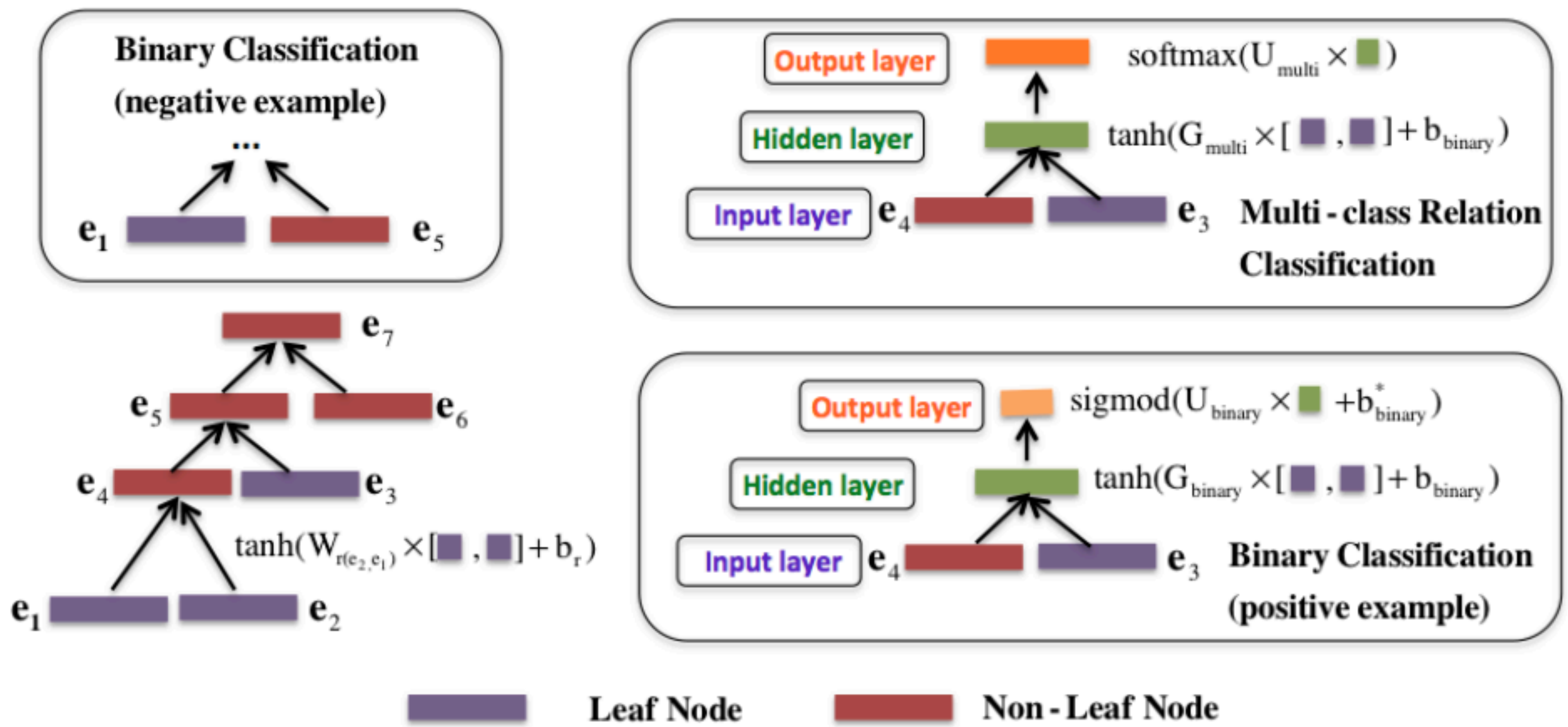


- Parse a piece of text into a relations between discourse units (EDUs).
- Researchers mainly used the Rhetorical Structure Theory (RST) formalism, which forms a tree of relations.

Example RST structures from Marcu (2000)

Recursive Deep Models for Discourse Parsing

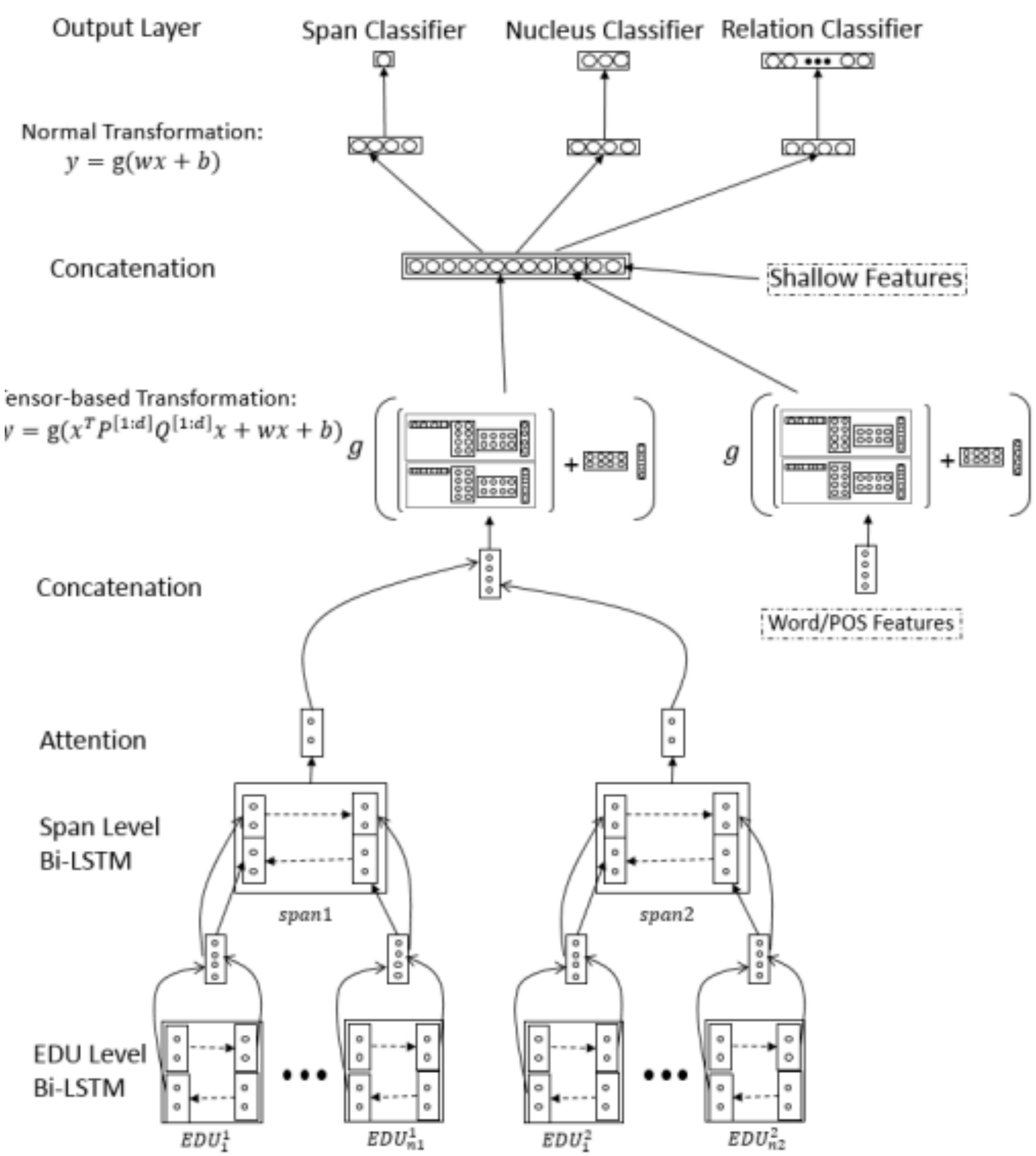
Li et.al (2014)



- 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

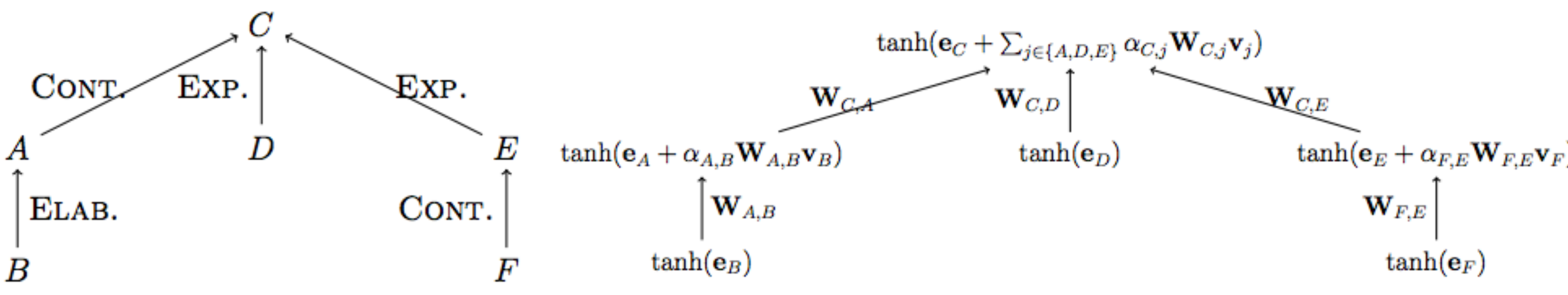
Li et.al (2016)



- 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

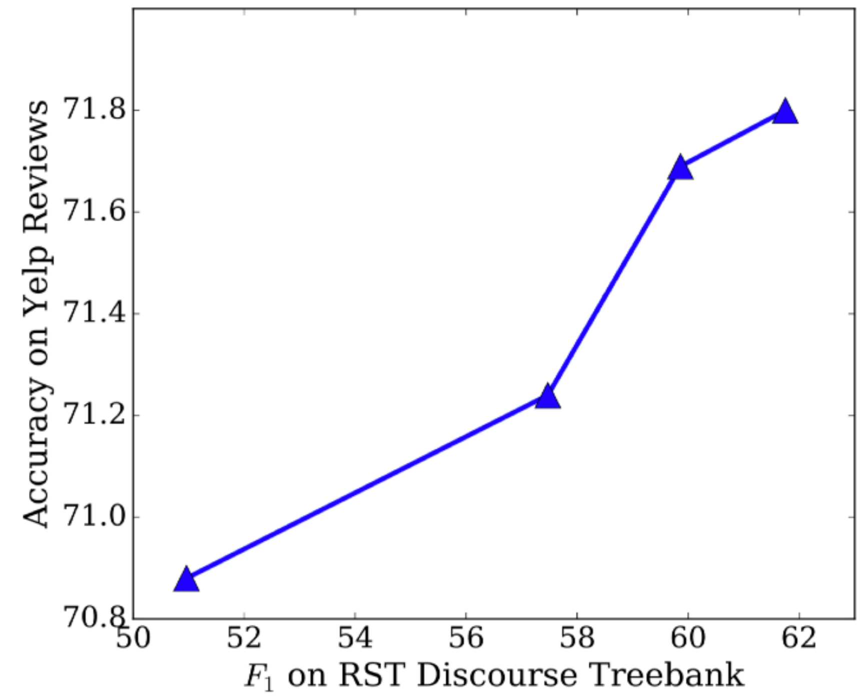
Uses of Discourse Structure in Neural Models

- 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?