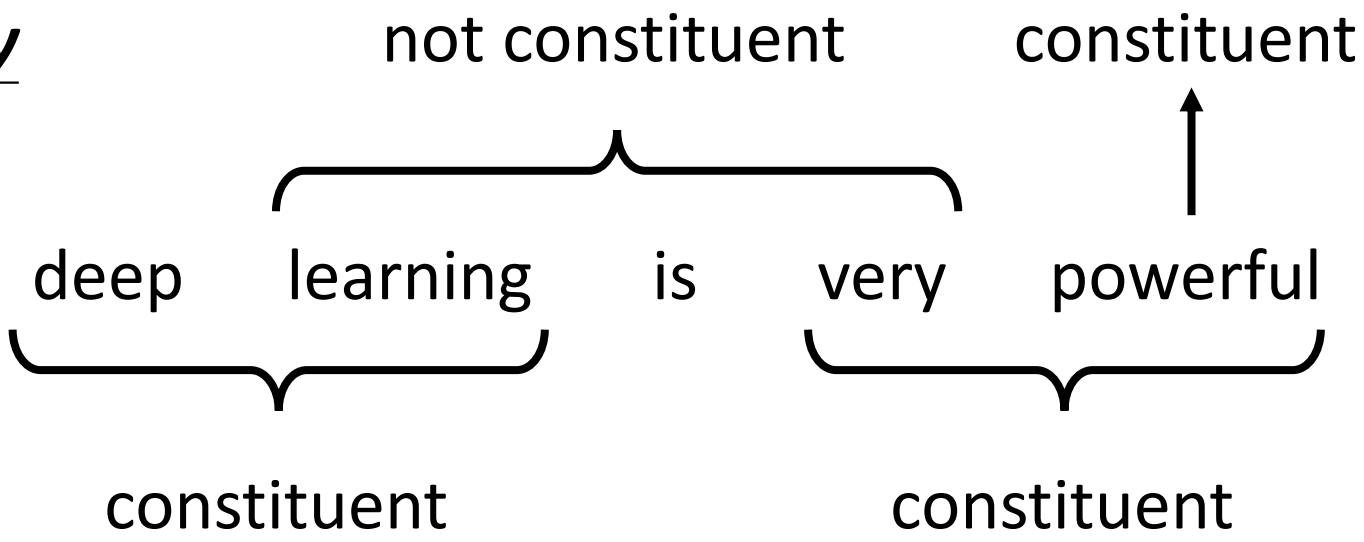

Dependency Parsing
Hung-yi Lee 李宏毅

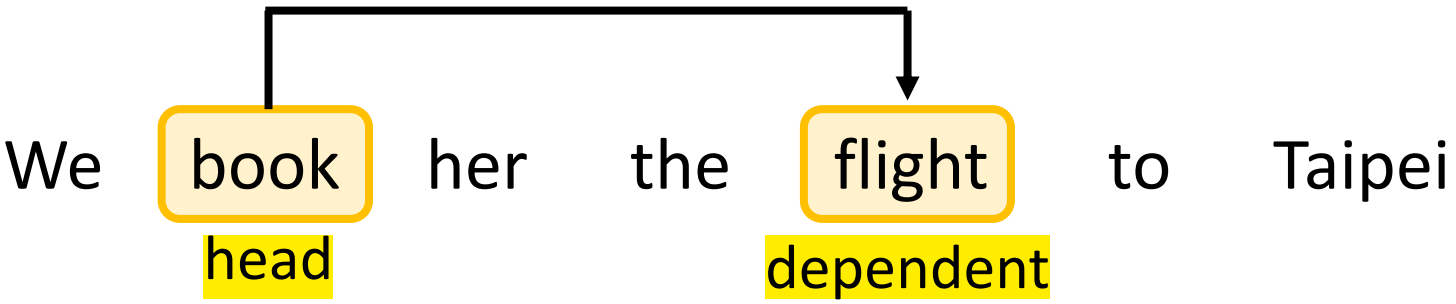
| | One Sequence | Multiple Sequences |
|---|---|---|
| One Class | Sentiment Classification<br>Stance Detection<br>Veracity Prediction<br>Intent Classification<br>Dialogue Policy | NLI<br>Search Engine<br>Relation Extraction |
| Class for each Token | POS tagging<br>Word segmentation<br>Extractive Summarization<br>Slotting Filling<br>NER | |
| Copy from Input | | Extractive QA |
| General Sequence | Abstractive Summarization<br>Translation<br>Grammar Correction<br>NLG | General QA<br>Chatbot<br>State Tracker<br>Task Oriented Dialogue |
| Other? | Parsing, Coreference Resolution | |

***Constituency Parsing***




***Dependency Parsing***

两个词汇之间的关系



| Clausal Argument Relations | Description |
|---|---|
| NSUBJ | Nominal subject |
| DOBJ | Direct object |
| IOBJ | Indirect object |
| CCOMP | Clausal complement |
| XCOMP | Open clausal complement |
| **Nominal Modifier Relations** | **Description** |
| NMOD | Nominal modifier |
| AMOD | Adjectival modifier |
| NUMMOD | Numeric modifier |
| APPOS | Appositional modifier |
| DET | Determiner |
| CASE | Prepositions, postpositions and other case marker |
| **Other Notable Relations** | **Description** |
| CONJ | Conjunct |
| CC | Coordinating conjunction |

**Figure 15.2** Selected dependency relations from the Universal Dependency set. (de Maeffe et al., 2014)

# Dependency Parsing

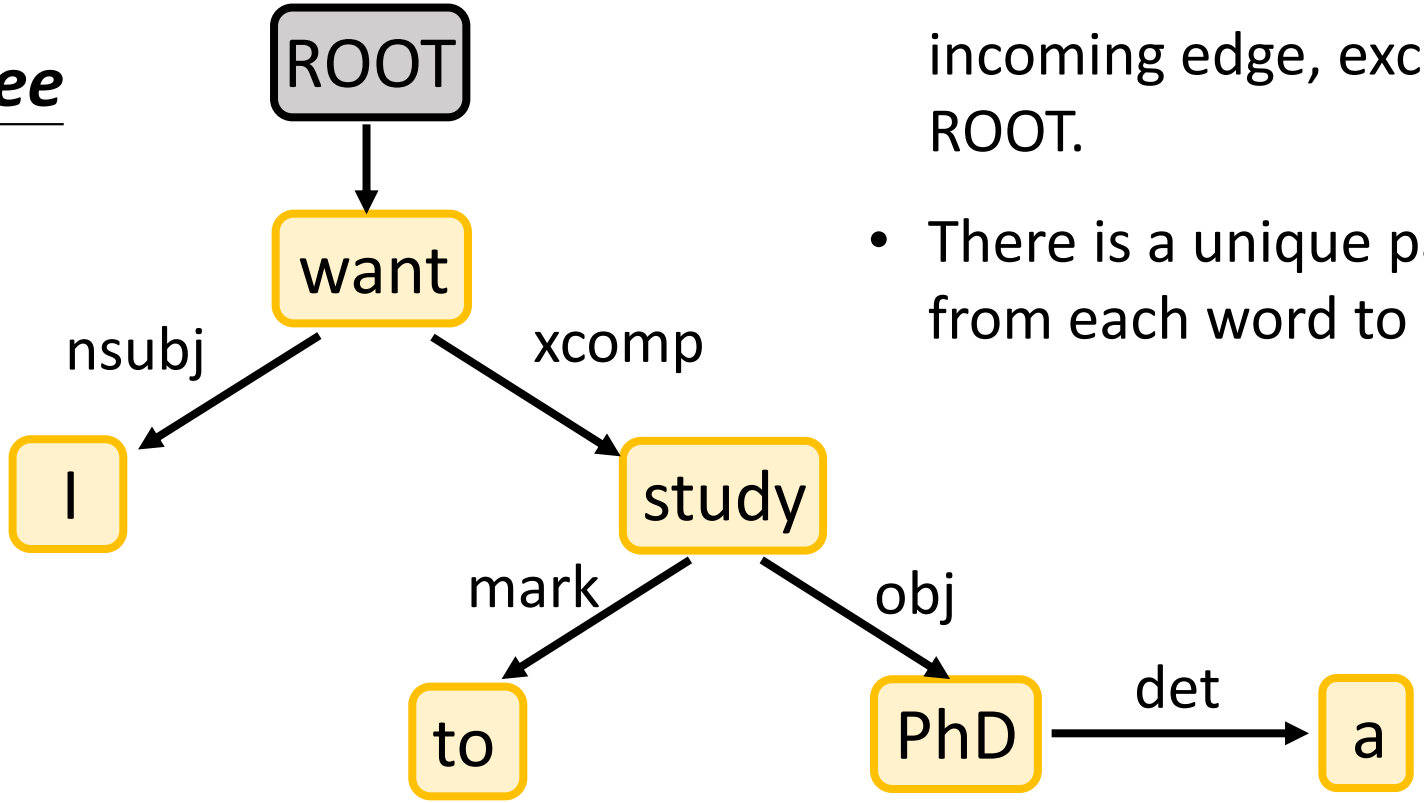
I want to study a PhD

Directed graph

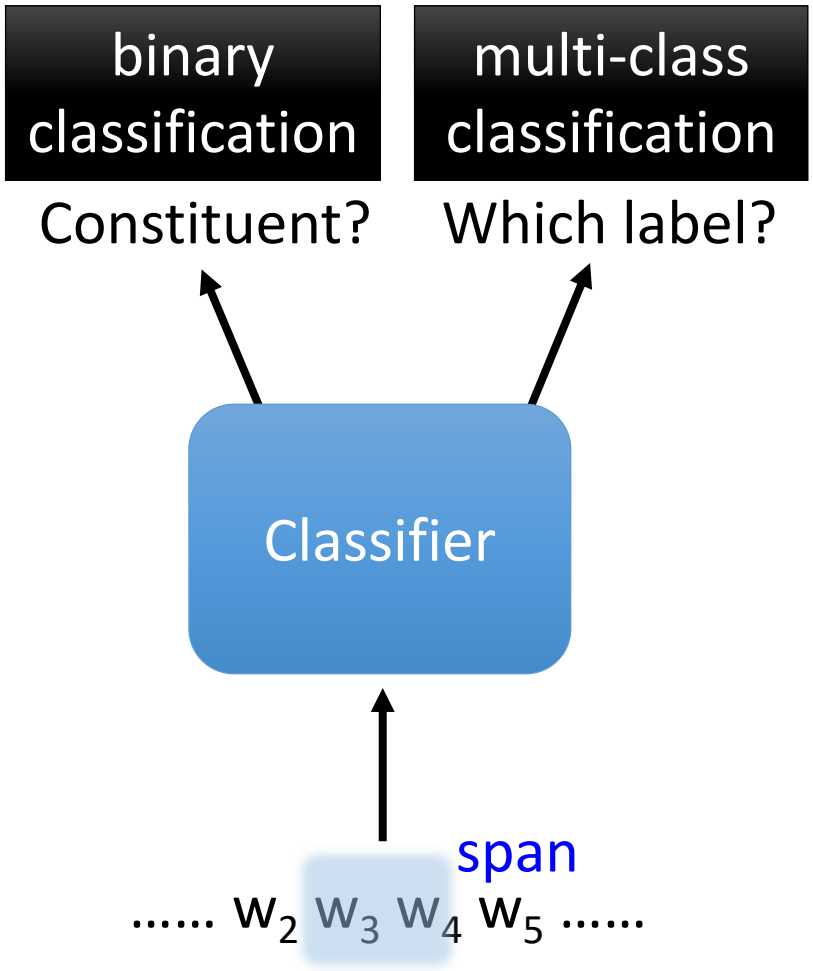
node → word

edge → relation

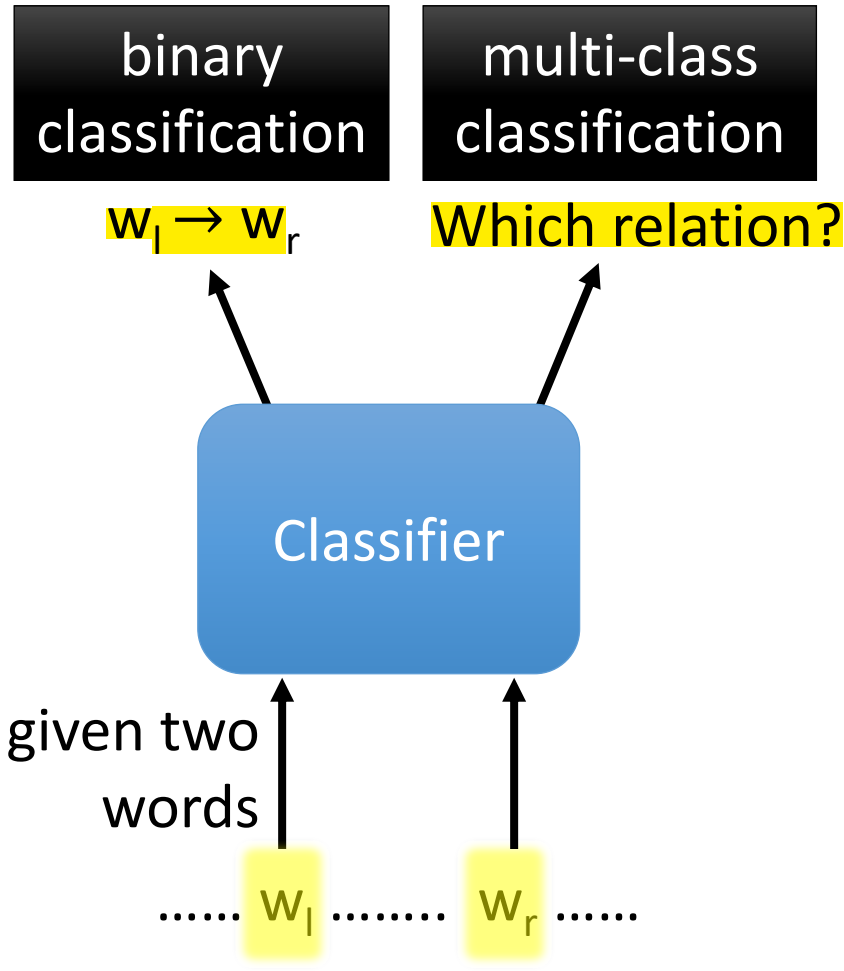
***Tree***




- All the words have one incoming edge, except ROOT.
- There is a unique path from each word to ROOT.



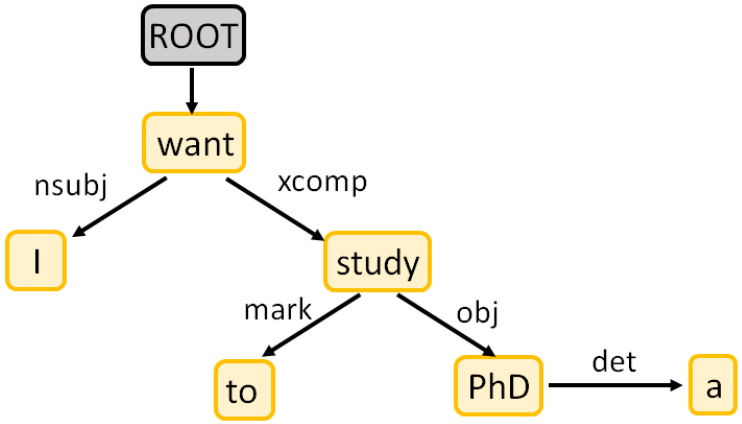

***Constituency Parsing***




***Dependency Parsing***

# Graph-based

Run the classifier at most $(N+1)^2$ times




YES

$w_l \rightarrow w_r$

Classifier

ROOT I want

NO

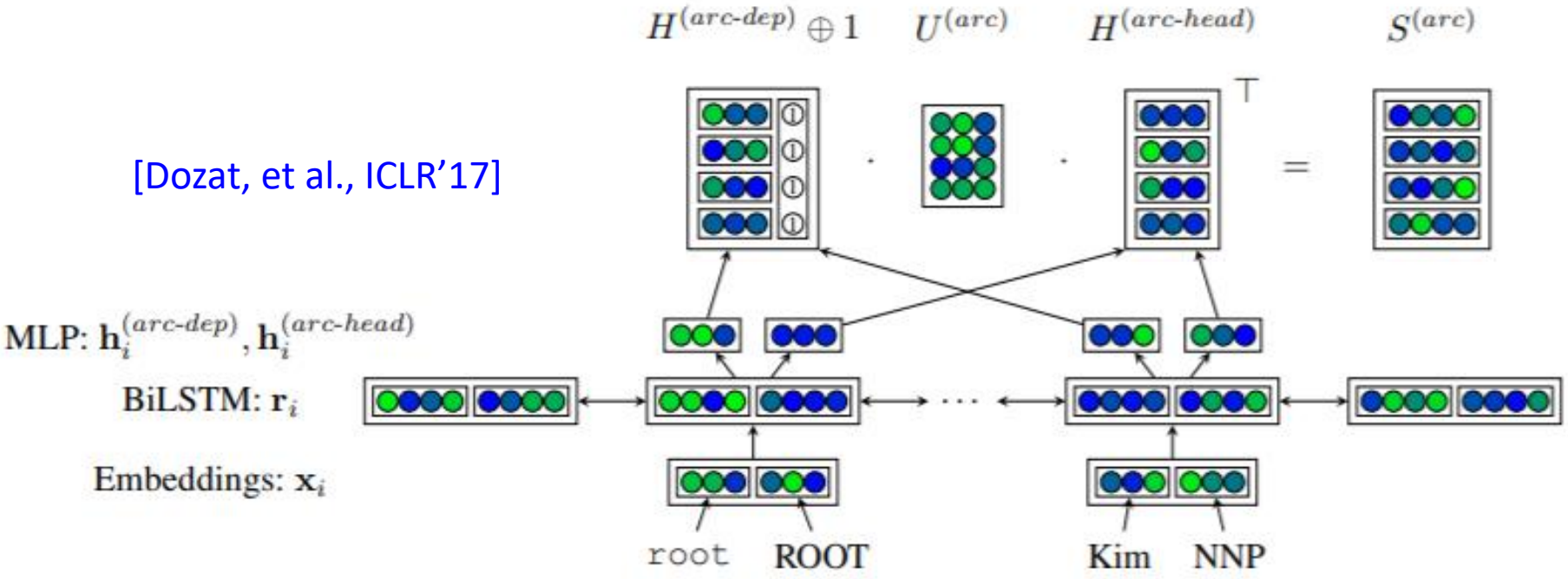
$w_l \rightarrow w_r$

Classifier

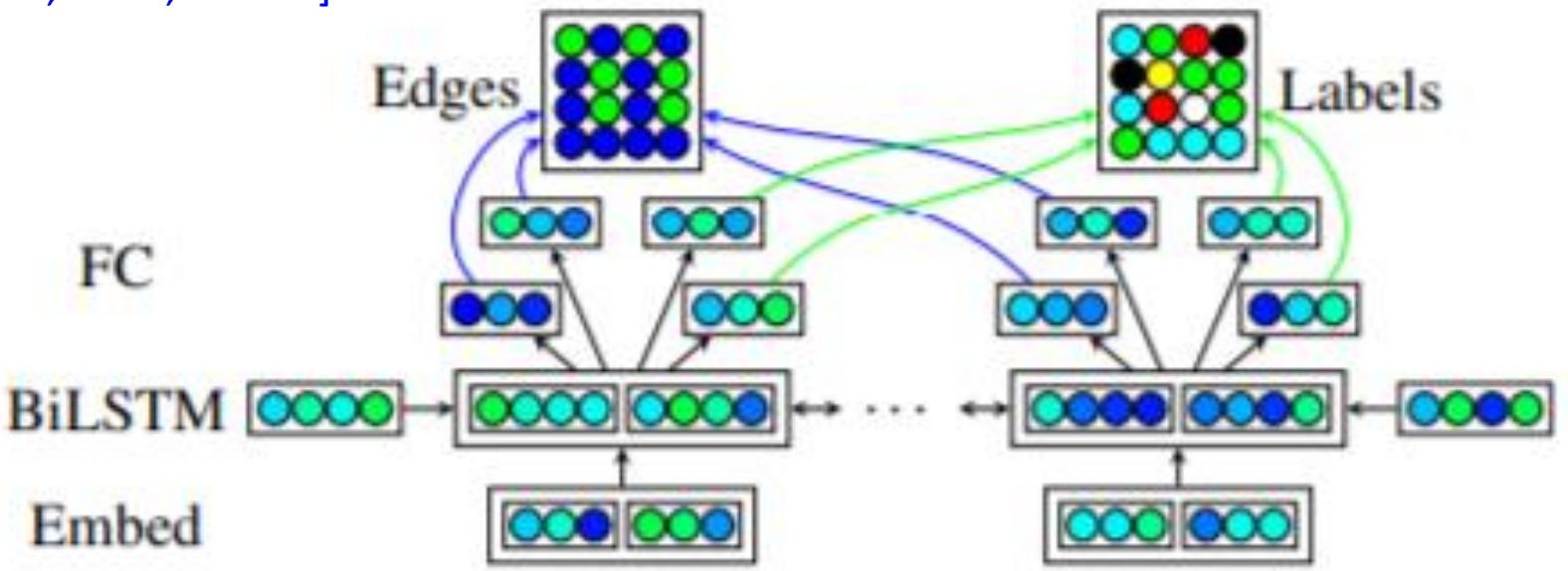
to study a PhD

N words

[Dozat, et al., ICLR'17]

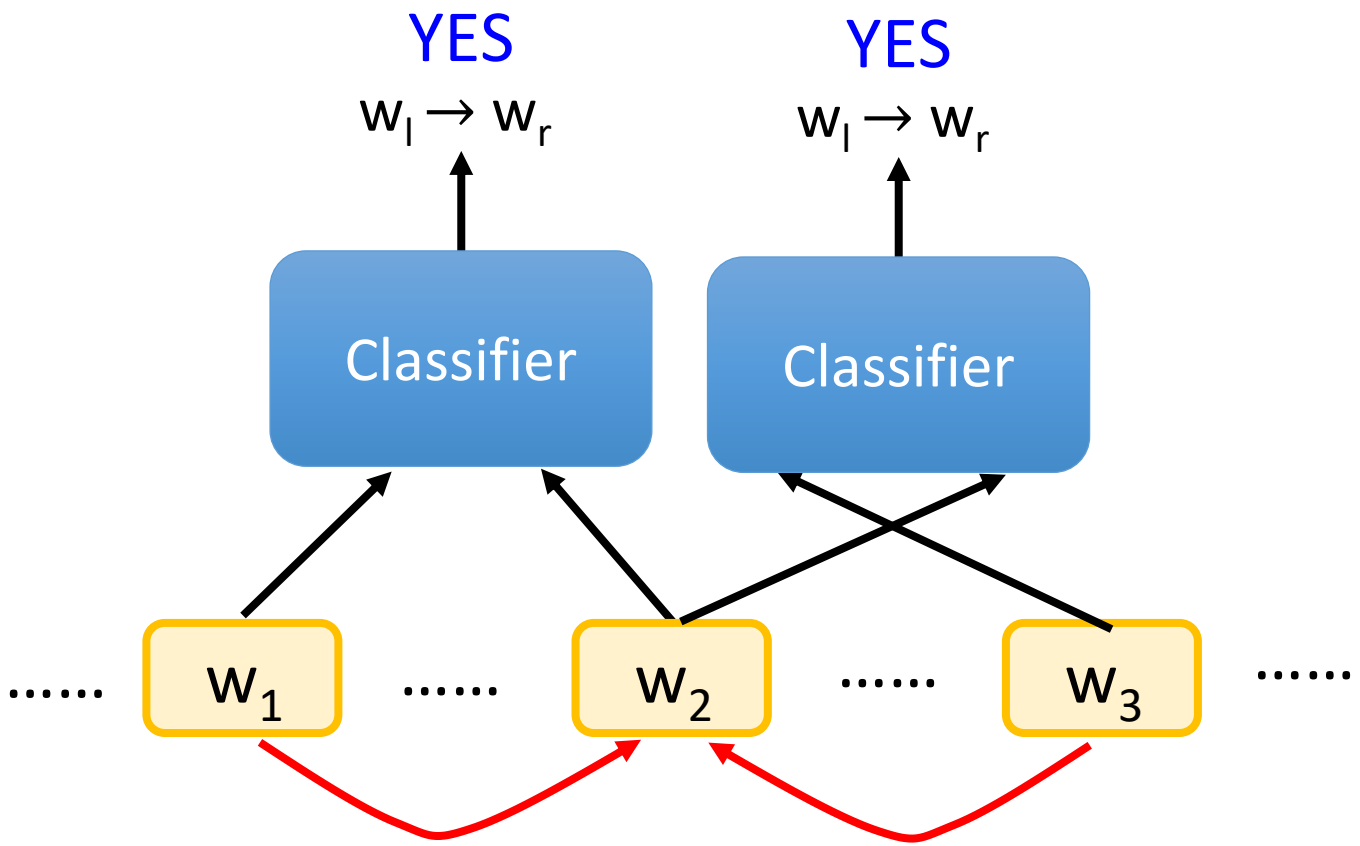



[Dozat, et al., ACL'18]

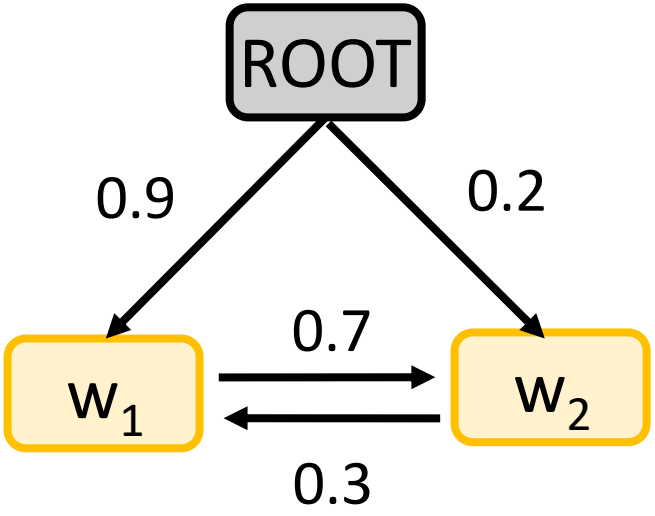


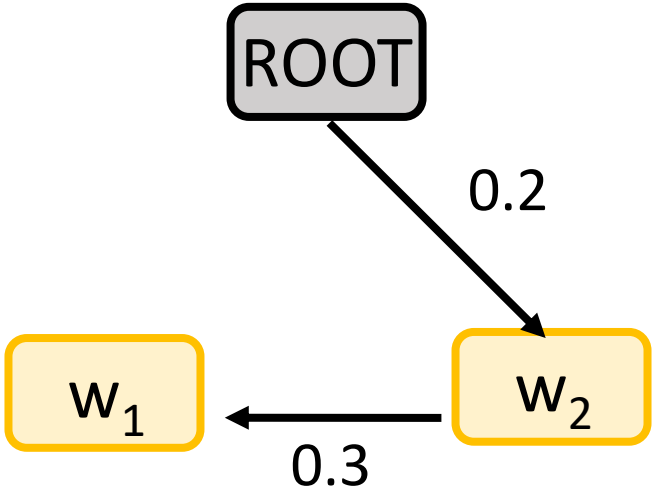
# Graph-based

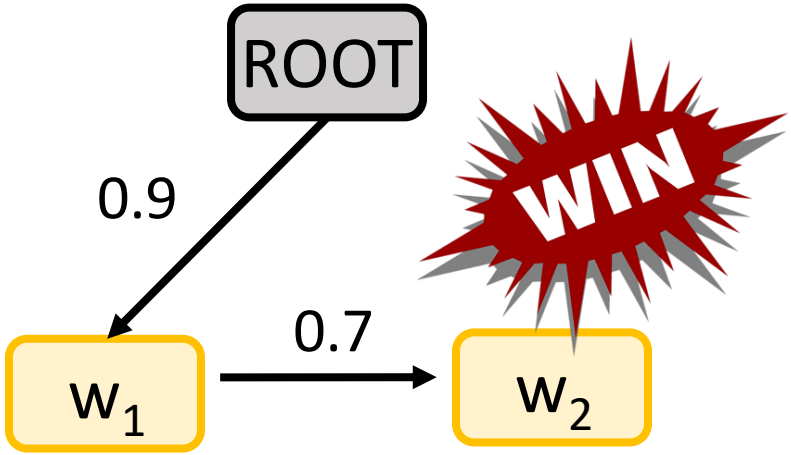
Contradiction!



# Maximum Spanning Tree



# Transition-based Approach

| Step | Stack | Word List | Action | Relation Added |
|---|---|---|---|---|
| 0 | [root] | [book, me, the, morning, flight] | SHIFT | |
| 1 | [root, book] | [me, the, morning, flight] | SHIFT | |
| 2 | [root, book, me] | [the, morning, flight] | RIGHTARC | (book → me) |
| 3 | [root, book] | [the, morning, flight] | SHIFT | |
| 4 | [root, book, the] | [morning, flight] | SHIFT | |
| 5 | [root, book, the, morning] | [flight] | SHIFT | |
| 6 | [root, book, the, morning, flight] | [] | LEFTARC | (morning ← flight) |
| 7 | [root, book, the, flight] | [] | LEFTARC | (the ← flight) |
| 8 | [root, book, flight] | [] | RIGHTARC | (book → flight) |
| 9 | [root, book] | [] | RIGHTARC | (root → book) |
| 10 | [root] | [] | Done | |

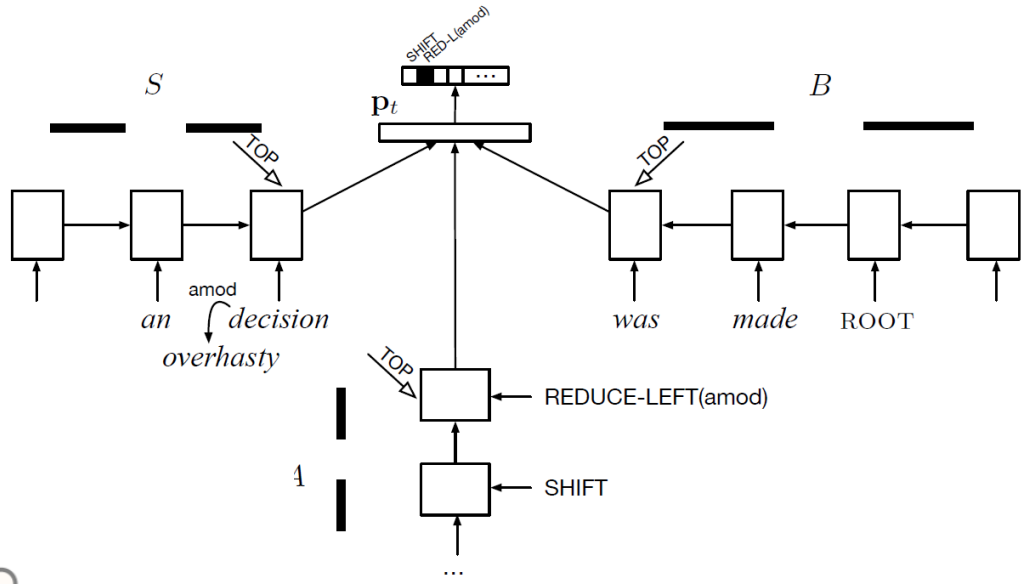
**Figure 15.7** Trace of a transition-based parse.

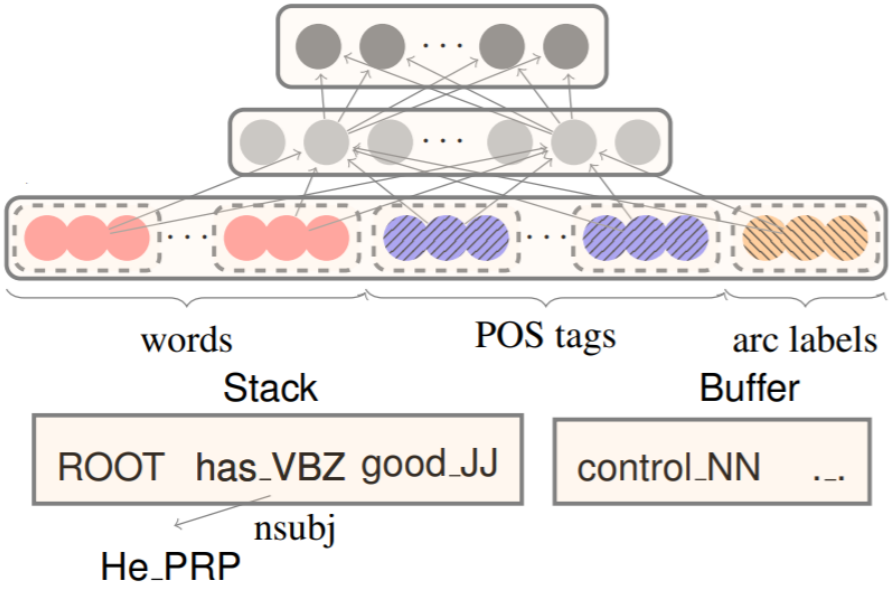
A stack, a buffer, some actions ……

We have learned similar approaches when talking about constituency parsing.

# Transition-based Approach

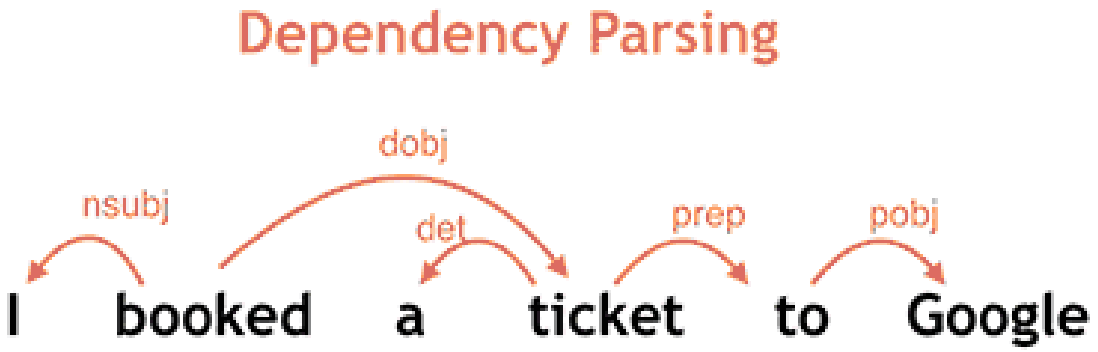
[Dyer, et al., ACL'15]



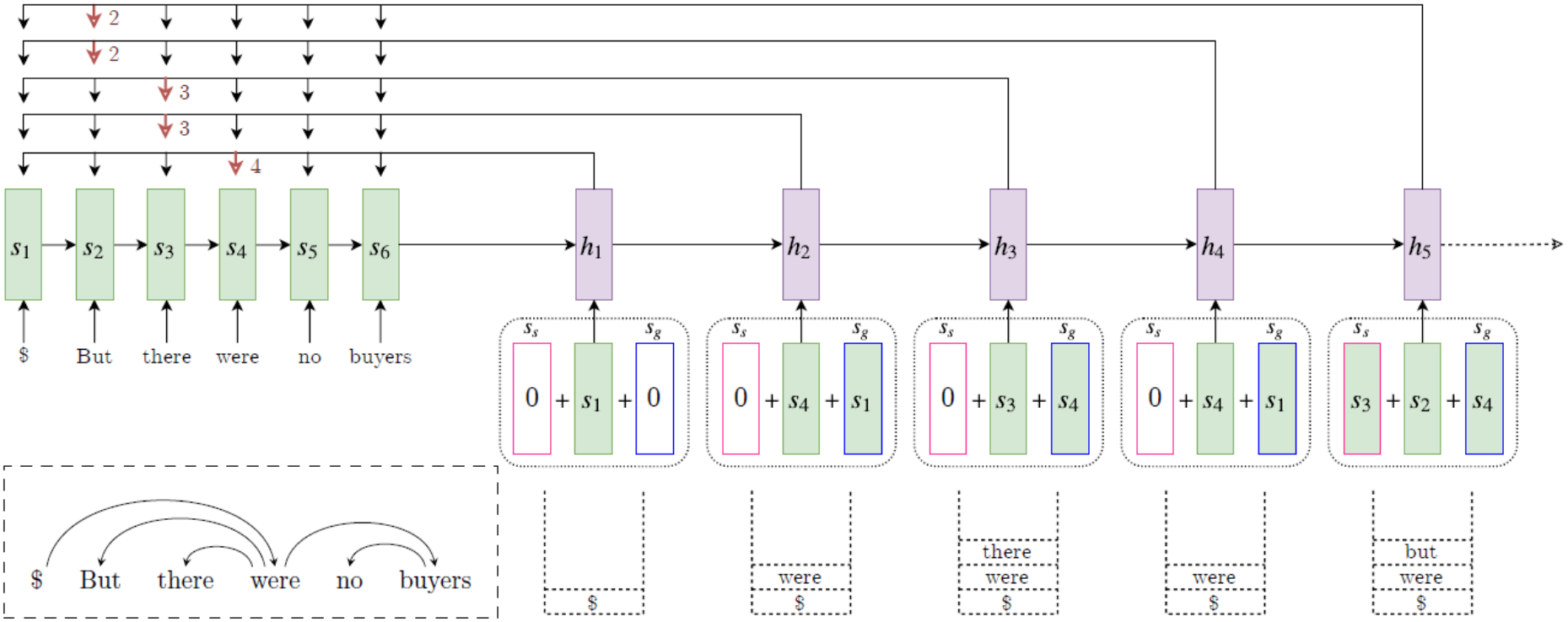

[Chen, et al., EMNLP'14]

# SyntaxNet [Andor, et al., ACL'16]




https://ai.googleblog.com/2016/05/announcing-syntaxnet-worlds-most.html

# Stack Pointer




[Ma, et al., ACL'18]

# Reference


- Danqi Chen, Christopher D. Manning, A Fast and Accurate Dependency Parser using Neural Networks, EMNLP, 2014
- Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, Noah A. Smith, Transition-Based Dependency Parsing with Stack Long Short-Term Memory, ACL, 2015
- Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov and Michael Collins, Globally Normalized Transition-Based Neural Networks, ACL, 2016
- Timothy Dozat, Christopher D. Manning, Deep Biaffine Attention for Neural Dependency Parsing, ICLR, 2017
- Timothy Dozat, Christopher D. Manning, Simpler but More Accurate Semantic Dependency Parsing, ACL, 2018
- Xuezhe Ma, Zecong Hu, Jingzhou Liu, Nanyun Peng, Graham Neubig, Eduard Hovy, Stack-Pointer Networks for Dependency Parsing, ACL, 2018