Stanford NLP

Evan Jaffe and Evan Kozliner

Some Notable Researchers

- Chris Manning
  - Statistical NLP, Natural Language Understanding and Deep Learning
- Dan Jurafsky
  - Natural Language Understanding, conversational speech & dialog and NLP for the social sciences
- Percy Liang
  - Semantic parsing, probabilistic models for NLP

Manning Article

- A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task
    - ACL 2016 Outstanding Paper Awards
    - Reading comprehension task
    - Google’s “Deep Mind” researchers built the data set based on CNN and Daily Mail
    - Question and answer dataset (based on bullet points)
    - Coreference chains replaced with markers (ensures the machine is “reading” the text and not just using a language mode to answer the question)
    - Small number of features such as the frequency of a potential answer
    - Classifiers examined: LSTM and conventional ensemble of decision trees
    - Accuracy around 75% (potentially the best possible due to the data prep)

Jurafsky Article

- Predicting the Rise and Fall of Scientific Topics from Trends in their Rhetorical Framing
    - Concept of rhetorical scholarly functions of *topics* (as opposed to sentences)
    - Examines the relationship between the “rhetorical role” of topics (for example if a topic is being used as a result) and the popularity of the articles they belong to
    - Interesting system involving several pieces:
        - Algorithm to tag the “rhetorical roles” of topics in articles where they are untagged, creates a tuple of topics to rhetorical roles
        - Use this and the popularity of the article at that time to create time series of the rhetorical functions and their popularity in that time
        - Cluster time series in unsupervised way
        - Classify time series with logistic regression based on their cluster

Improving Coreference Resolution by Learning Entity-Level Representations

Clark and Manning, 2016 ACL

Identity Coreference: structured prediction that links mentions that have the same referent

The city council denied the demonstrators a permit because...
... they feared violence.
... they advocated violence.
(Winograd 1972)

Improving Coreference Resolution by Learning Entity-Level Representations

Clark and Manning, 2016 ACL

Identity Coreference: structured prediction that links mentions that have the same referent

The city council denied the demonstrators a permit because...
... they feared violence.
... they advocated violence.
(Winograd 1972)

Their Approach

Easy-first agglomerative clustering

Start state: all mentions in their own cluster

Merge clusters until final partition reached.

Deciding whether to merge or not depends on the score of the resulting cluster vs. the score of the mention being a singleton.

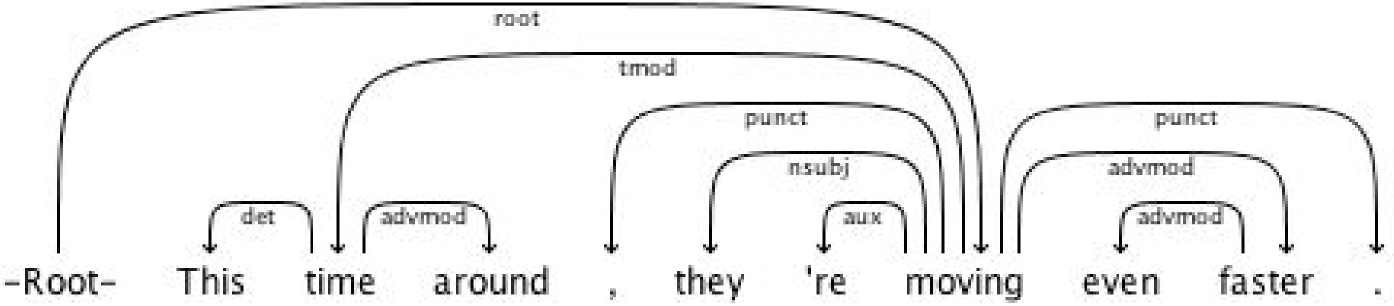
Using the scoring functions s_c and s_{NA}, we define a policy network π that assigns a probability distribution over $U(x)$ as follows:

$$\pi(\textsc{Merge}[c_m, c] | x) \propto e^{s_c(c_m, c)}$$

$$\pi(\textsc{Pass} | x) \propto e^{s_{\text{NA}}(m)}$$

A Fast and Accurate Parser Using Neural Networks

Chen and Manning, 2014 EMNLP



Universal Dependencies, Marie-Catherine de Marneffe (OSU LING)

Transition-based parsing: buffer, stack, linear scan

Contrast with bottom-up parsing: speed, theoretical issues

Transition-based Parsing

Stack: partial parse, including words currently being processed

Buffer: words yet to be processed

Parser applies transitions to its state until its buffer is empty and the dependency graph is completed. Initial state: all words in order on the buffer, with a single dummy ROOT node on the stack.

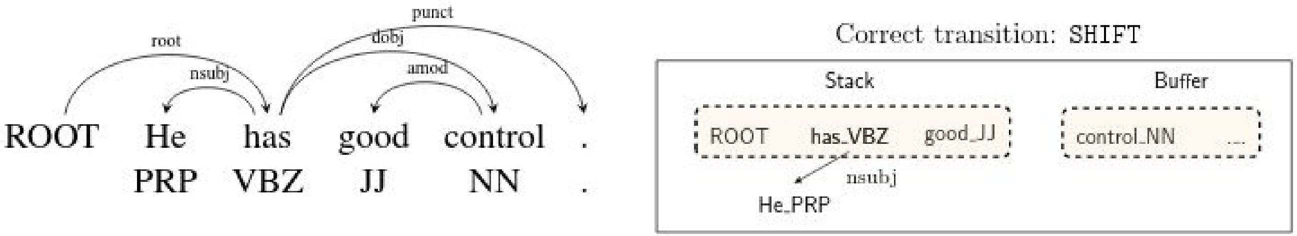
Transitions:

LEFT-ARC: marks the second item on the stack as a dependent of the first item, and removes the second item from the stack (if the stack contains at least two items).

RIGHT-ARC: marks the first item on the stack as a dependent of the second item, and removes the first item from the stack (if the stack contains at least two items).

SHIFT: removes a word from the buffer and pushes it onto the stack (if the buffer is not empty).

Transition-based Parsing



Transition	Stack	Buffer	A
	[ROOT]	[He has good control .]	$\emptyset$
SHIFT	[ROOT He]	[has good control .]	
SHIFT	[ROOT He has]	[good control .]	
LEFT-ARC(nsubj)	[ROOT has]	[good control .]	$A\cup$ nsubj(has,He)
SHIFT	[ROOT has good]	[control .]	
SHIFT	[ROOT has good control]	[.]	
LEFT-ARC(amod)	[ROOT has control]	[.]	$A\cup$amod(control,good)
RIGHT-ARC(dobj)	[ROOT has]	[.]	$A\cup$ dobj(has,control)
...	...	...	...
RIGHT-ARC(root)	[ROOT]	[]	$A\cup$ root(ROOT,has)

Figure 1: An example of transition-based dependency parsing. Above left: a desired dependency tree, above right: an intermediate configuration, bottom: a transition sequence of the arc-standard system.