Unsupervised grammar induction

Unsupervised Grammar Induction?

- Start with raw text, learn syntactic structure
- Some have argued that learning syntax from positive data alone is impossible:
  - Gold, 1967: Non-identifiability in the limit
  - Chomsky, 1980 The poverty of the stimulus
- Many others have felt it should be possible:
  - Lari and Young, 1990
  - Carroll and Charniak, 1992
  - Alex Clark, 2001
  - Mark Paskin, 2001
  - … and many more, but it hasn't worked well

American Structuralists

- Methods of automatically doing linguistic induction (if stated in a somewhat informal, non-computational manner) were the basis of the proposed 'discovery procedures' argued for by structural linguists in the 1950s:
  - "Many linguists felt that the procedures had been so well worked out that computers could take over the drudgery of linguistic analysis. The time was near at hand when all one would have to do would be to punch the data into the computer and out would come the grammar!" (Newmeyer 1986)

Supervised Learning

- Systems duplicate correct analyses from training data
- Hand-annotation of data
  - Time-consuming
  - Expensive
  - Hard to adapt for new purposes (tasks, languages, domains, etc)
  - Corpus availability drives research, not tasks!
- Example: Penn Treebank
  - 50K Sentences
  - Hand-parsed over several years

Unsupervised Learning

- Systems take raw data and automatically detect patterns
- Why unsupervised learning?
  - More data than annotation
  - Insights into machine learning, clustering
  - Kids learn some aspects of language entirely without supervision
- Here: unsupervised learning
  - Work purely from the forms of the utterances
  - Syntactic learnability: Neither assume nor exploit prior meaning or grounding [*cf.* Feldman *et al.*]

Gold (1967) vs. Horning (1969)

- Gold: no superfinite class of languages (e.g., regular or context-free languages, etc.) is learnable without negative examples.
  - Certain conditions: nearly arbitrary sequence of examples; only constraint is that no sentence may be withheld from the learner indefinitely.
- Still regularly cited as bedrock for innatist linguistics
  - "The Poverty of the Stimulus" (Chomsky 1965)
- Responses suggested by Gold:
  - Subtle, covert negative evidence ← Some recent claims
  - Innate knowledge shrinks language class ← Chomsky
  - Assumption about presentation of examples is too general ← e.g., probabilistic language model

Gold (1967) vs. Horning (1969)

- If texts are generated by some stochastic process, then we can take advantage of how often something occurs to drive language acquisition
  - In particular, as time goes by without us having seen something, we have more evidence that it either doesn't happen or happens very, very rarely in the language
  - Implicit negative evidence: you can't withhold common stuff
- Horning: stochastic context-free languages are learnable* from only positive examples.
- But Horning's proof is enumerative, rather than providing a plausible grammar learning method

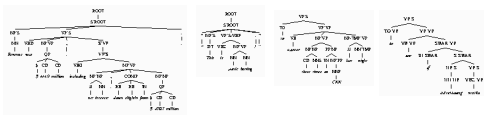
(See, e.g., Rohde and Plaut 1999 for discussion)

Poverty of the stimulus?

- There's a lot of stimulus: Baayen estimate of 200 million words by adulthood
  - c. 1,000,000 sentences a year
- Much more sign of other poverties:
  - Poverty of imagination
  - Poverty of model building
  - Poverty of understanding of (machine) learning among linguists
- At any rate, an obvious way to investigate this question is empirically....

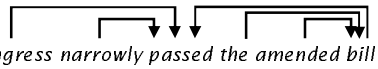
NLP

- This is also an engineering problem
- Because of the very modest success of grammar induction systems, practical work has largely abandoned induction, and is dependent on large hand-built resources
- Means that we're *barely learning*. Instead, getting human to externalize knowledge.
- It's scarcely machine learning at all....
- Machine learning should be looking at feature/representation induction!



Idea: Lexical Affinity Models

- Words select other words on syntactic grounds



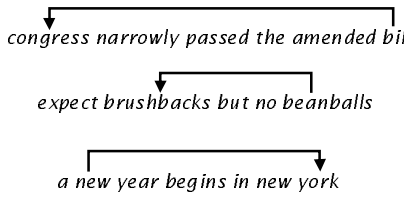
congress narrowly passed the amended bill

- Link up pairs with high mutual information
  - [Yuret, 1998, MIT PhD]: Greedy linkage
  - [Paskin, 2001, NIPS]: Iterative re-estimation with EM
- Evaluation: compare linked pairs to a gold standard

Method	Accuracy
Paskin, 2001	39.7

Problem: Non-Syntactic Affinity

- Mutual information between words does not necessarily indicate syntactic selection.



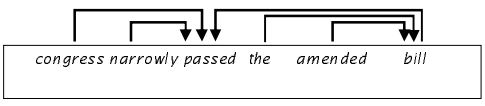
congress narrowly passed the amended bill

expect brushbacks but no beanballs

a new year begins in new york

Idea: Word Classes

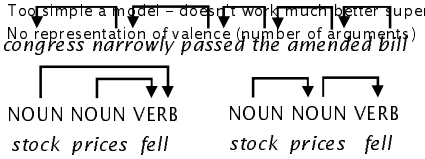
- Individual words like congress are entwined with semantic facts about the world.
- Syntactic classes, like NOUN and ADVERB are bleached of word-specific semantics.
- Automatic word classes more likely to look like DAYS-OF-WEEK or PERSON-NAME.
- We could build dependency models over word classes. [*cf.* Carroll and Charniak, 1992]



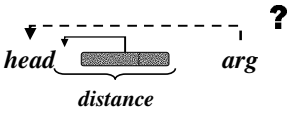
Problems: Word Class Models

Random	41.7
Carroll and Charniak, 92	44.7

- Issues:
  - Too simple a model – doesn't work much better supervised
  - No representation of valence (number of arguments)



Bias: Using more sophisticated dependency representations



	Classes?	Distance	Local Factor
Paskin 01	✗	✗	$P(a \mid h)$
Carroll & Charniak 92	✓	✗	$P(c(a) \mid c(h))$
Our Model (DMV)	✓	✓	$P(c(a) \mid c(h), d)$

Adjacent Words	55.9
Our Model (DMV)	63.6

Grammar Structure search: Greedy-Merge

- Agglomerative clustering of part of speech sequences
  - Similarity by normalized divergence of contexted sequence
- At each step merge two closest sequences.
- Merging can create recursion.

DT NN DT JJ NN	➡	z1 → DT NN z1 → DT JJ NN	Create a category
NN JJ NN	➡	NN → JJ NN z1 → DT NN z1 → DT [JJ NN] ➡ z1 → DT [NN]	1. Add to a category 2. Re-analyze rules

Learned Grammars (NP)

N-bar or zero determiner NP
zNN → NN | NNS
zNN → JJ zNN
zNN → zNN zNN

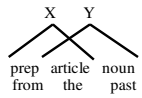
Proper NP
zNNP → NNP | NNPS
zNNP → zNNP zNNP

NP with determiner
zNP → DT zNN
zNP → PRP$ zNN

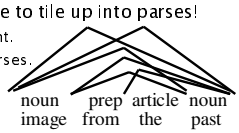
Categories have been hand re-labeled based on closest treebank category.

Problems with Structure Search

- Greedy systems make local constituency decisions.
  - Chunk prep article → X or
  - Chunk article noun → Y ?
  - Decide heuristically!
  - Many heuristics are often right, but
  - Early errors stay around



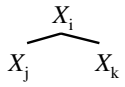
- In the end, constituents have to tile up into parses!
  - Global no-overlap constraint.
  - Requires comparing full parses.
  - Not heuristic.



Idea: Learn PCFGs with EM (Parameter Search)

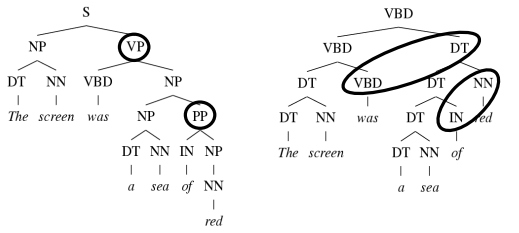
- Classic experiments on learning PCFGs with Expectation-Maximization [Lari and Young, 1990]

$$\{ X_1 , X_2 \ldots X_n \}$$



  - Full binary grammar over n symbols
  - Parse randomly at first
  - Re-estimate rule probabilities off parses
  - Repeat
- Their conclusion: it doesn't work at all!

Example Parse: PCFG EM



Treebank Parse

DEP-PCFG Parse

Linguistic constituency

- EM assumptions/greediness may not work:
  - The important tests for linguistic constituency emphasize external distribution
  - Not internal constituency (e.g., Radford 1988)
- Why? Look at Noun Phrases:

you	<s> ___ saw me
the man	<s> He saw ___ </s>
a cat with a limp	the elephant sat on __
three black dogs	My gosh! ___ </s>
Don't look alike	*All* appear in these (and other) contexts

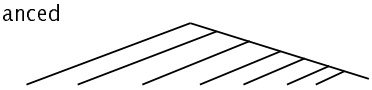
Other Approaches

- Some recent work in learning constituency:
  - [Adrians, 99] Language grammars aren't general PCFGs
  - [Clark, 01] Mutual-information filters detect constituents, then an MDL-guided search assembles them
  - [van Zaanen, 00] Finds low edit-distance sentence pairs and extracts their differences
  - GB/Minimalism No results that I've seen
- Evaluation: fraction of nodes in gold trees correctly posited in proposed trees (unlabeled recall)

Adriaans, 1999	16.8
Clark, 2001	34.6
van Zaanen, 2000	35.6

Right-Branching Baseline

- English trees tend to be right-branching, not balanced



they were unwilling to agree to new terms

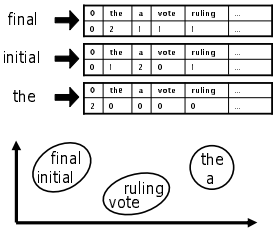
- A simple (English-specific) baseline is to choose the right-branching structure for each sentence

van Zaanen, 00	35.6
Right-Branch	46.4

Inspiration: Distributional Clustering for capturing linguistic classes

- Item tokens occur in specific contexts.
- Item types are identified with their context distribution.
- Distance between items defined by distance between their contexts.
- Used for clustering.

... the final vote ...
... a final ruling ...
... the final scheduled ...



Learning Language Distributionally

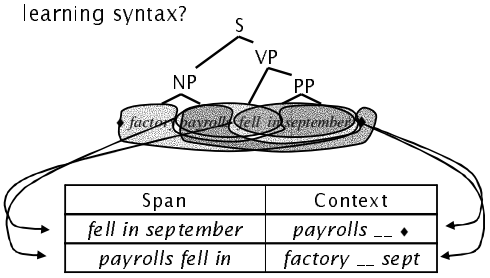
Item	Context	Learned	Past Work
words	left, right word	parts-of-speech	Finch et. al. 95, Schütze 95, Clark 00
words	content words	semantic fields	Pereira et. al. 93, Dagan et. al. 97

... it generally "pretty much" works:

- parts-of-speech are partially semantic
- constituent types require some filtering

Idea: Distributional Syntax?

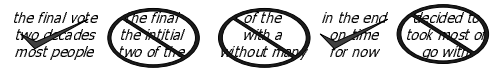
- Can we use distributional clustering for learning syntax?



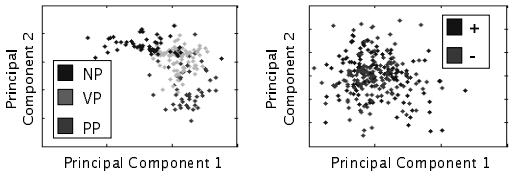
Span	Context
fell in september	*payrolls __ ♦*
payrolls fell in	*factory __ sept*

Problem: Identifying Constituents

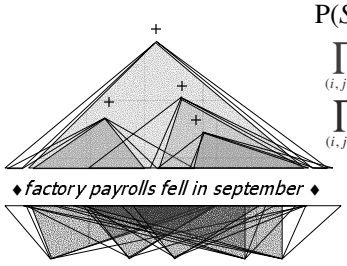
Distributional classes are easy to find…



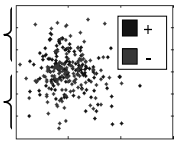
… but figuring out which are constituents is hard.



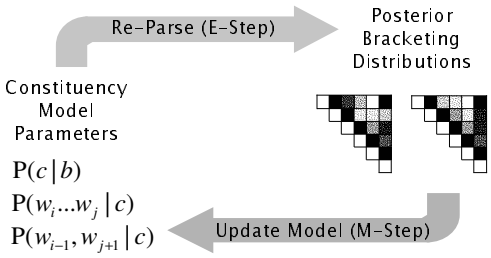
Constituent-Context Model (CCM)



$$P(S|T) = \prod_{(i,j)\in T} \begin{cases} P(\phi_{ij}|+)P(\chi_{ij}|+) \\ P(\blacklozenge\_\_\blacklozenge|+) \end{cases} \prod_{(i,j)\notin T} \begin{cases} P(\phi_{ij}|-)P(\chi_{ij}|-) \\ P(\blacklozenge\_\_\ fell|+) \end{cases}$$



Parameter Estimation with EM



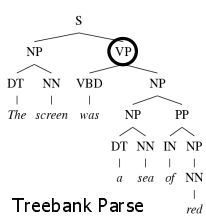
$P(c|b)$

$P(w_i \ldots w_j | c)$

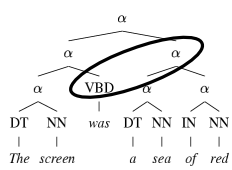
$P(w_{i-1}, w_{j+1} | c)$

Results: Constituency

Right-Branch	70.0	



Treebank Parse



CCM Parse

Spectrum of Systematic Errors

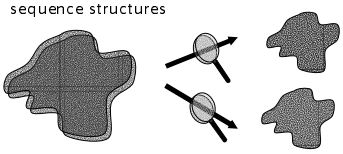
CCM analysis better ⟷ Treebank analysis better

Analysis	Inside NPs	Possesives	Verb groups
CCM	*the [lazy cat]*	*John ['s cat]*	*[will be] there*
Treebank	*the lazy cat*	*[John 's] cat*	*will [be there]*
CCM Right?	Yes	Maybe	No

But the worst errors are the non-systematic ones! (~25%)

Combining These Models

- General case: [Klein and Manning, 03]
  - Multiple scoring functions over a single family of tree or sequence structures



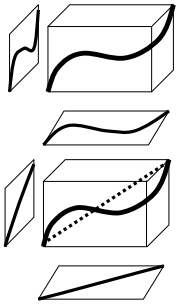
  - Each function scores some sub-projection
  - We want to find the best structure according to the sum or product of the individual functions
  - Can do A* search with bounds given by components

Factored A* Estimates

- If w factors over projections $\{\pi_i\}$, then for a path T:

$$w(T) = \sum_i w_i(\pi_i(T))$$

- Factored scores have a natural A* estimate:



Combining the two models

Dependency Evaluation

Random	45.6	
DMV	62.7	
CCM + DMV	64.7	

Constituency Evaluation

Random	39.4	
CCM	81.0	
CCM + DMV	88.0	

!

- *Supervised* PCFG constituency recall is at 92.8
- Qualitative improvements
  - Subject-verb groups gone, modifier placement improved

Crosslinguistic applicability of the learning algorithm

	Constituency Evaluation
English (7422 sentences)	
Random Baseline	39.4
CCM+DMV	88.0
German (2175 sentences)	
Random Baseline	49.6
CCM+DMV	89.7
Chinese (2473 sentences)	
Random Baseline	35.5
CCM+DMV	46.7

Maybe grammar induction is possible after all?

That's it folks!

- Reminders:
  - Get those final projects done!
    - *Must* be submitted by next Friday
  - Mini-presentations in class next Wednesday