New Results in Parsing

Eugene Charniak

Brown Laboratory for Linguistic
Information Processing

BLLIP

Joint Work with Mark Johnson and David McClosky

New Parsing Results

I Very Old Parsing Results (1990-2004)

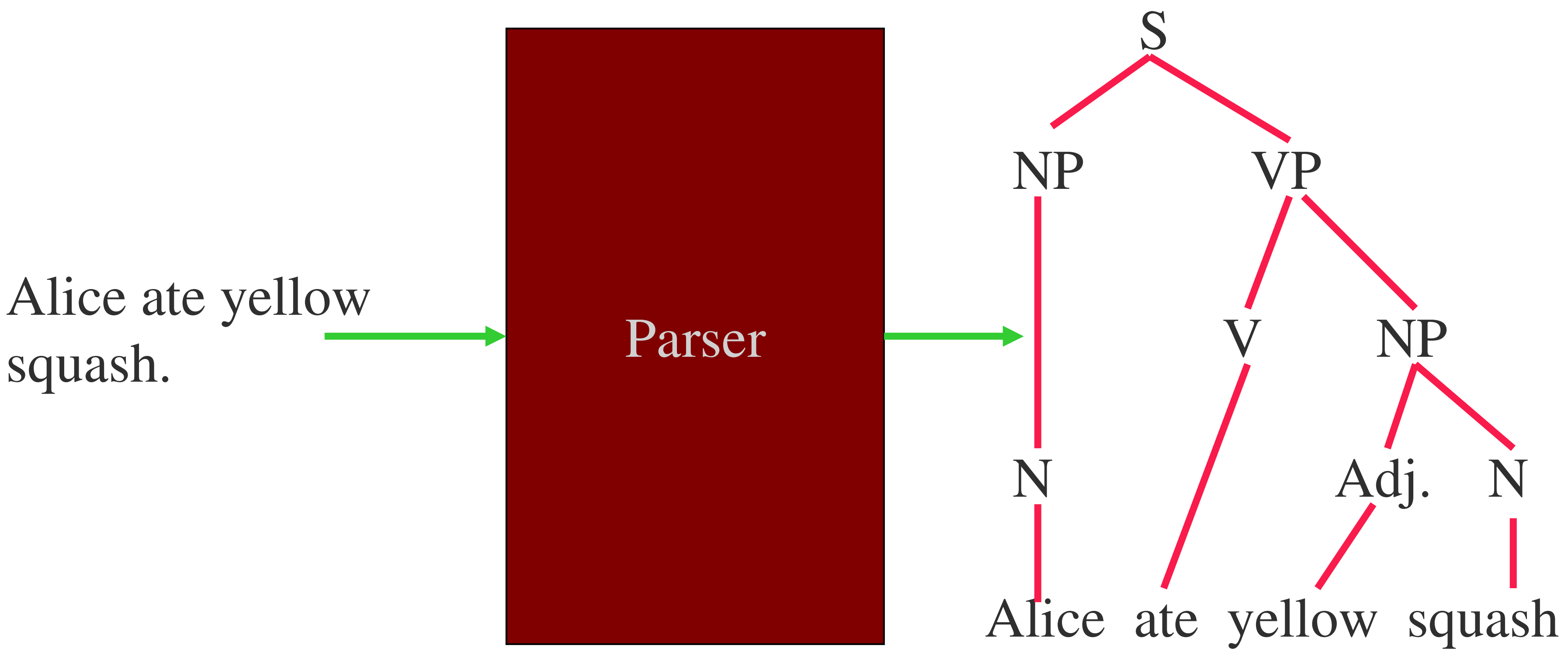
II More Recent Old Parsing Results (2004-2005)

III Learning from Unlabeled Data

IV Beyond the Wall-Street Journal

V Future Work

Parsing



Alice ate yellow squash.

Parser

S

NP

VP

V

NP

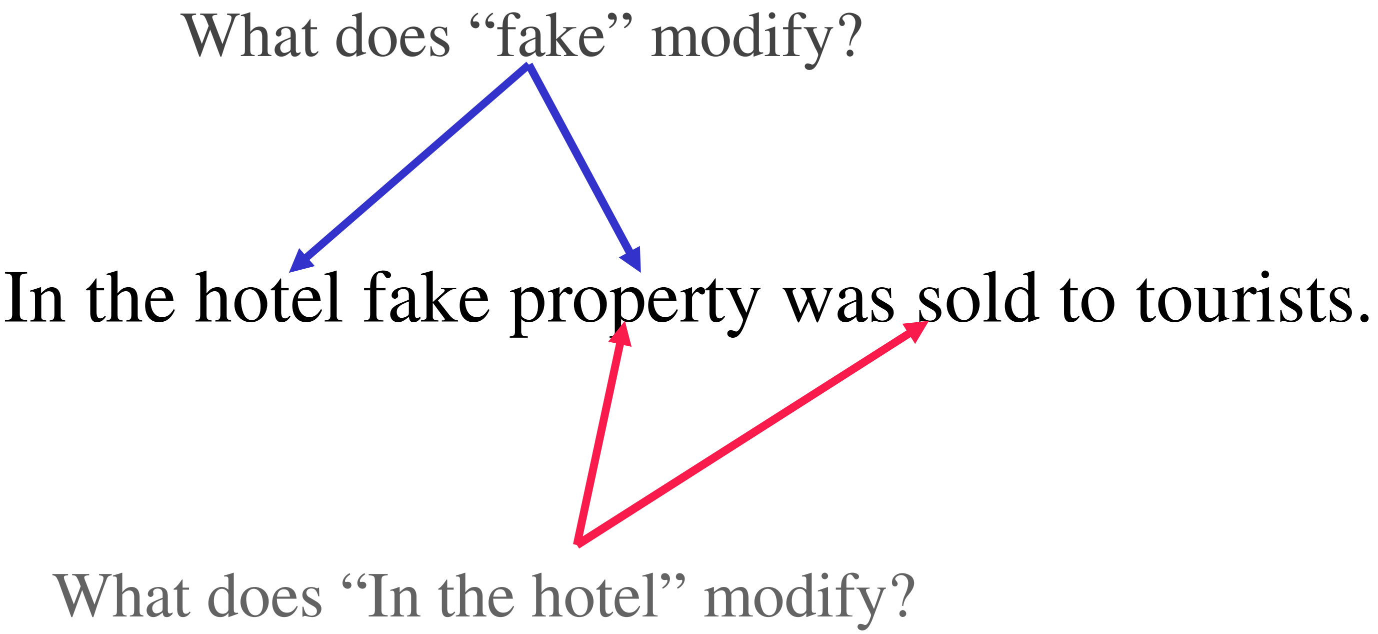
N

Adj.

N

Alice ate yellow squash

The Importance of Parsing

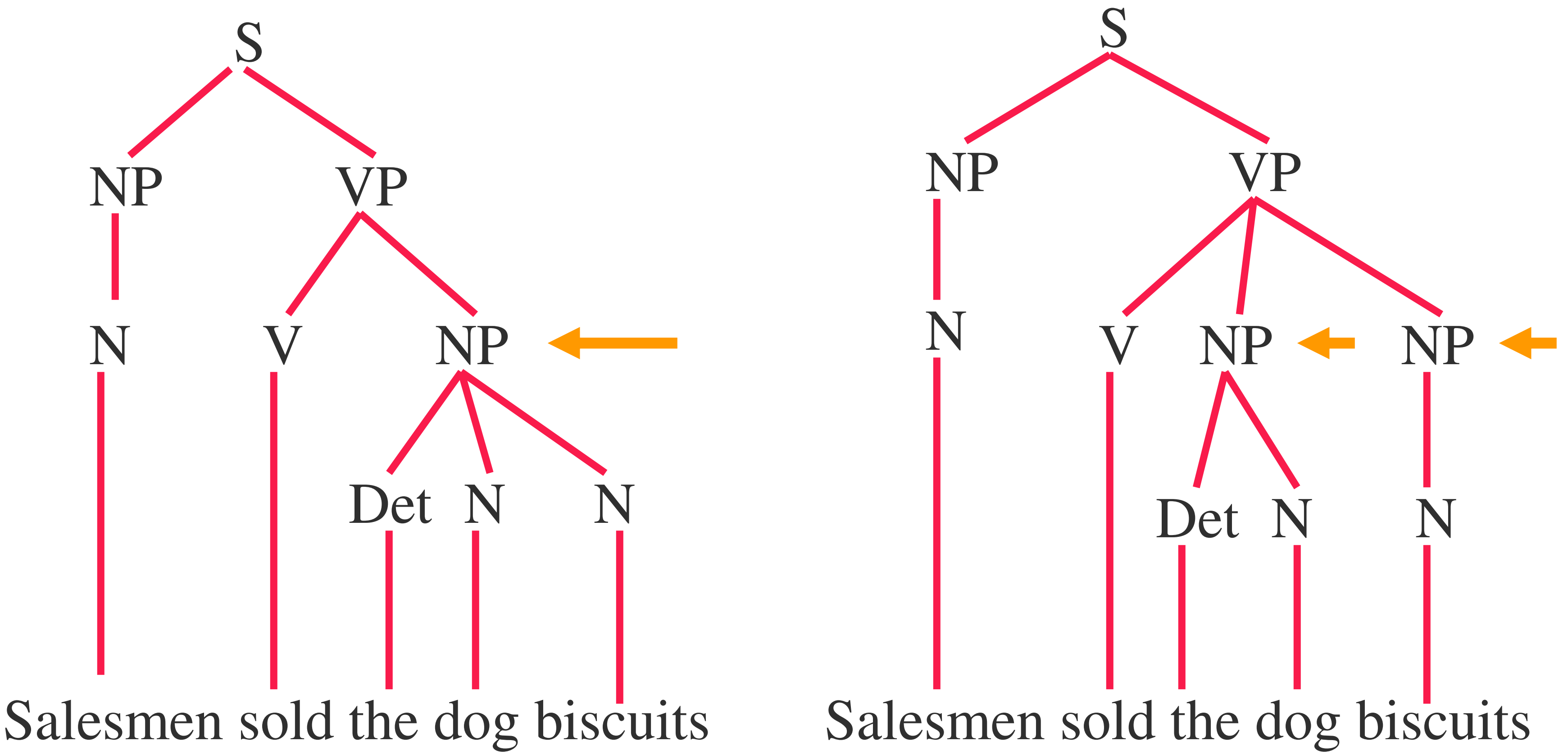


What does “fake” modify?

In the hotel fake property was sold to tourists.

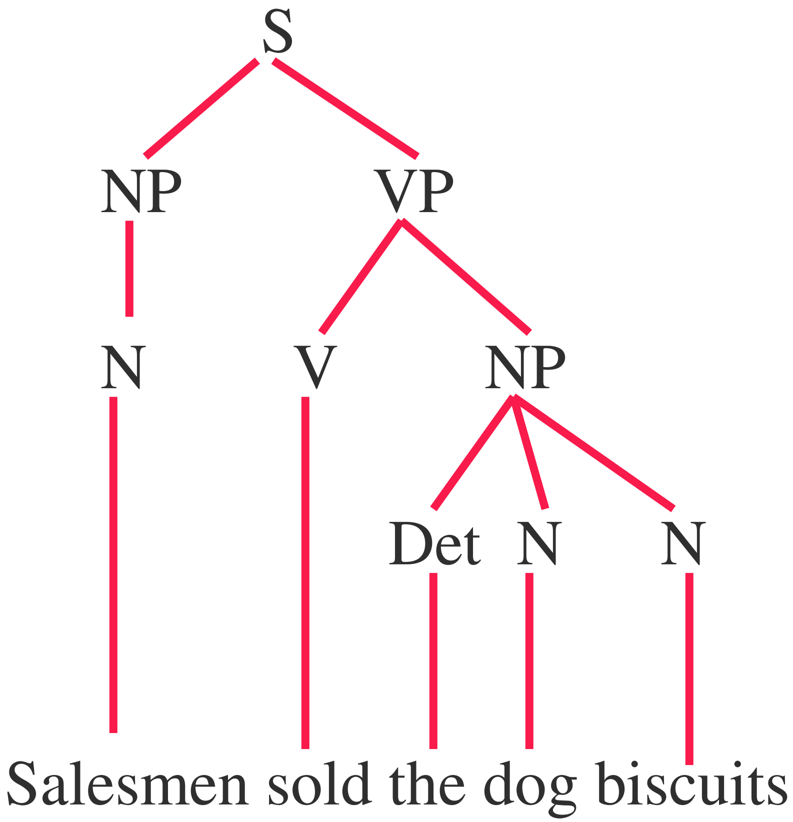
What does “In the hotel” modify?

Ambiguity

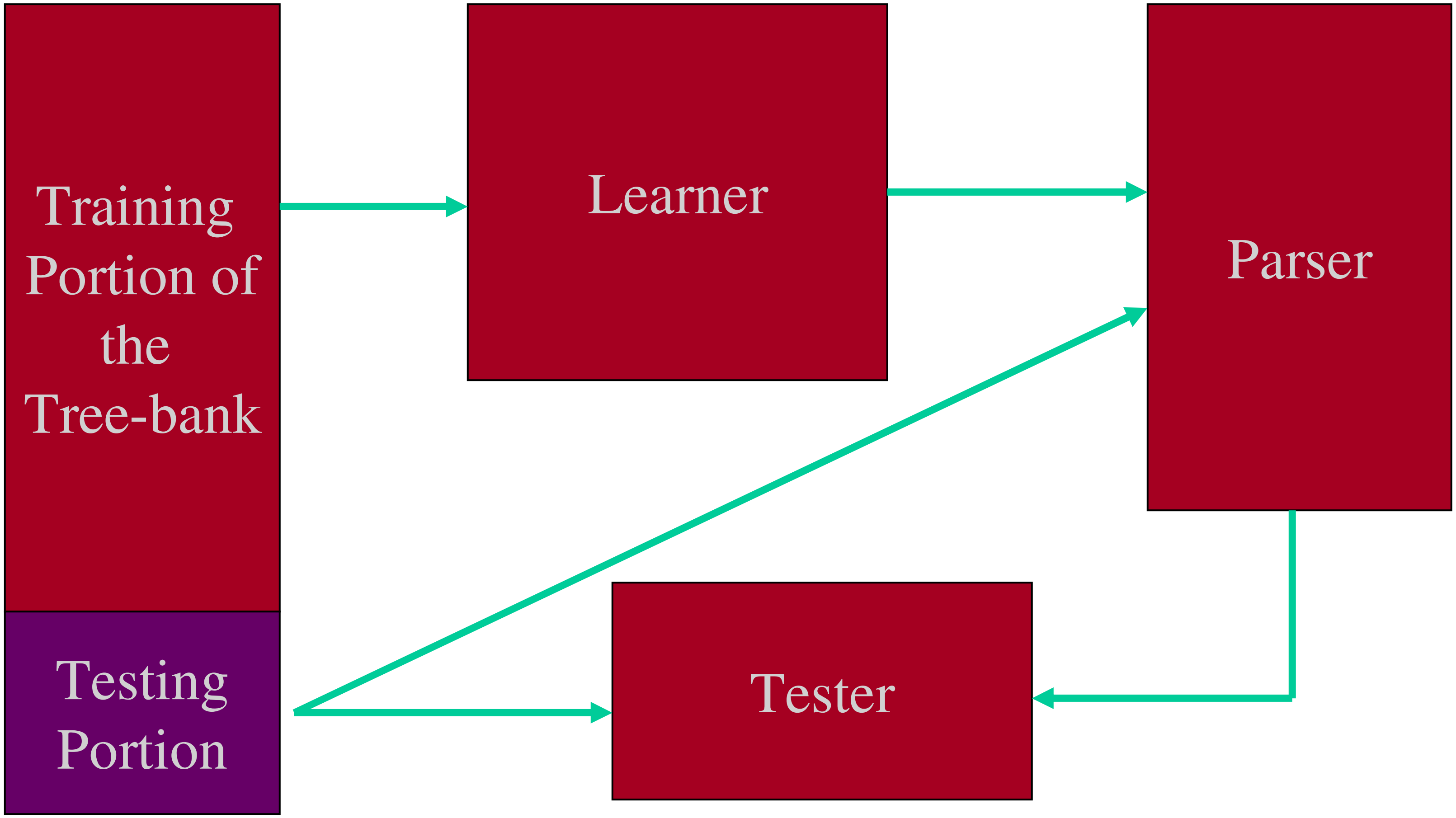

S
NP
VP
N
V
NP
Det
N
N
Salesmen sold the dog biscuits
S
NP
VP
N
V
NP
NP
Det
N
N
Salesmen sold the dog biscuits

Probabilistic Context-free Grammars (PCFGs)

S → NP VP	1.0
VP → V NP	0.5
VP → V NP NP	0.5
NP → Det N	0.5
NP → Det N N	0.5
N → salespeople	0.3
N → dog	0.4
N → biscuits	0.3
V → sold	1.0



The Basic Paradigm



Training Portion of the Tree-bank

Testing Portion

Learner

Parser

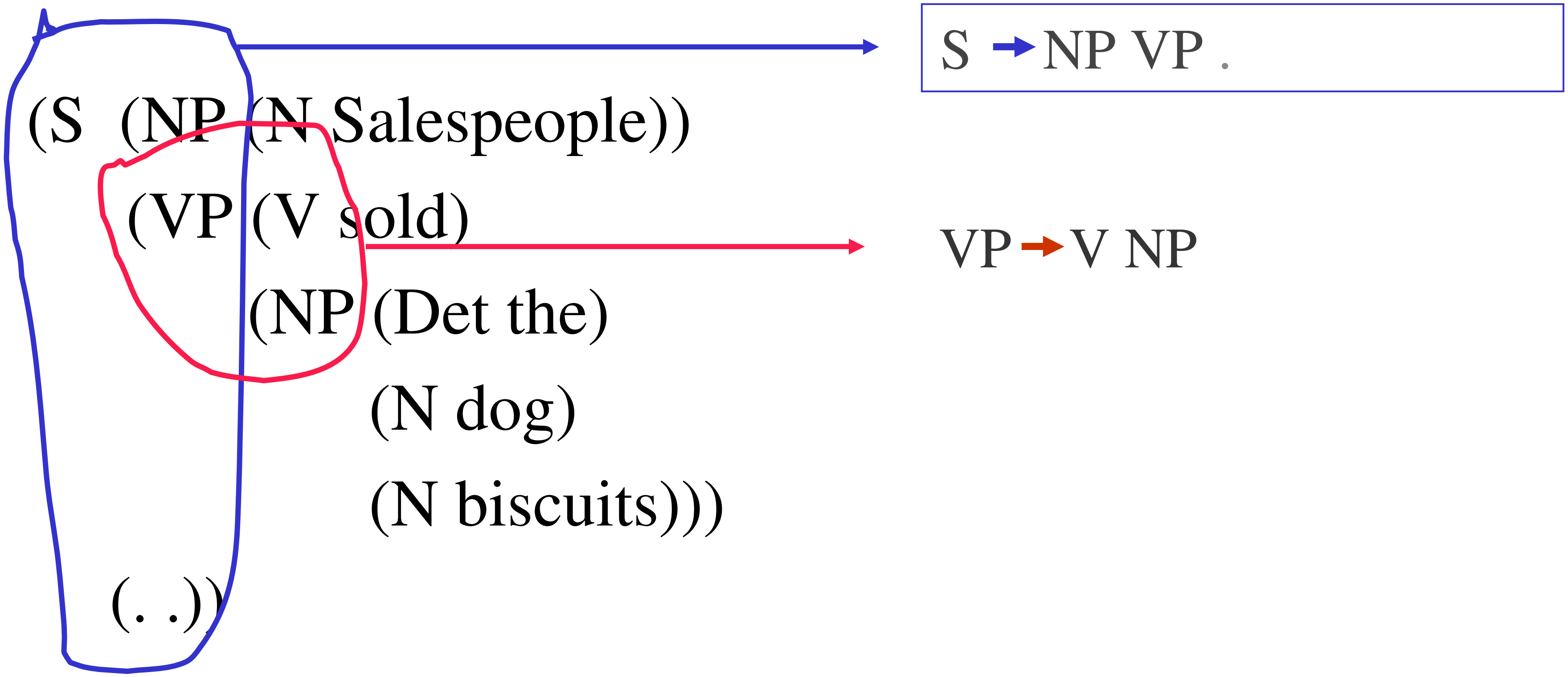
Tester

The Penn Wall Street Journal Tree-bank

- About one million words.
- Average sentence length is 23 words and punctuation.

> In an Oct. 19 review of “The Misanthrope” at Chicago’s Goodman Theatre (“Revitalized Classics Take the Stage in Windy City,” Leisure & Arts), the role of Celimene, played by Kim Cattrall, was mistakenly attributed to Christina Hagg.

"Learning" a PCFG from a Tree-Bank



```
(S  (NP (N Salespeople))
     (VP (V sold)
          (NP (Det the)
               (N dog)
               (N biscuits)))
     (. .))
```

S → NP VP .

VP → V NP

Producing a Single "Best" Parse

- The parser finds the most probable parse tree π given the sentence (***s***)

$$\begin{aligned}\arg\max_{\pi} p(\pi \mid s) &= \arg\max_{\pi} p(\pi, s) \quad \longleftarrow \\ &= \arg\max_{\pi} p(\pi) \quad \longleftarrow\end{aligned}$$

- For a PCFG we have the following, where c varies over the constituents in the tree π :

$$p(\pi) = \prod_{c \in \pi} p(\text{rule}(c)) \quad \longleftarrow$$

Evaluating Parsing Accuracy

- Few sentences are assigned a completely correct parse by any currently existing parsers. Rather, evaluation is done in terms of the percentage of correct constituents.

[label, start, finish]

- A constituent is a triple, all of which must be in the true parse for the constituent to be marked correct.

Evaluating Constituent Accuracy

Let C be the number of correct constituents produced by the parser over the test set, M be the total number of constituents produced, and N be the total in the correct version

Precision = C/M

Recall = C/N

It is possible to artificially inflate either one by itself.

I will typically give the harmonic mean (called the "f-measure"

Parsing Results

Method	Prec/Rec
PCFG	75%
PCFG + simple tuning	78%

Lexicalized Parsing

To do better, it is necessary to condition probabilities on the actual words of the sentence. This makes the probabilities much tighter:

$$p(\mathrm{VP} \rightarrow \mathrm{V\ NP\ NP}) = 0.00151$$

$$p(\mathrm{VP} \rightarrow \mathrm{V\ NP\ NP} \mid \mathrm{said}) = 0.00001$$

$$p(\mathrm{VP} \rightarrow \mathrm{V\ NP\ NP} \mid \mathrm{gave}) = 0.01980$$

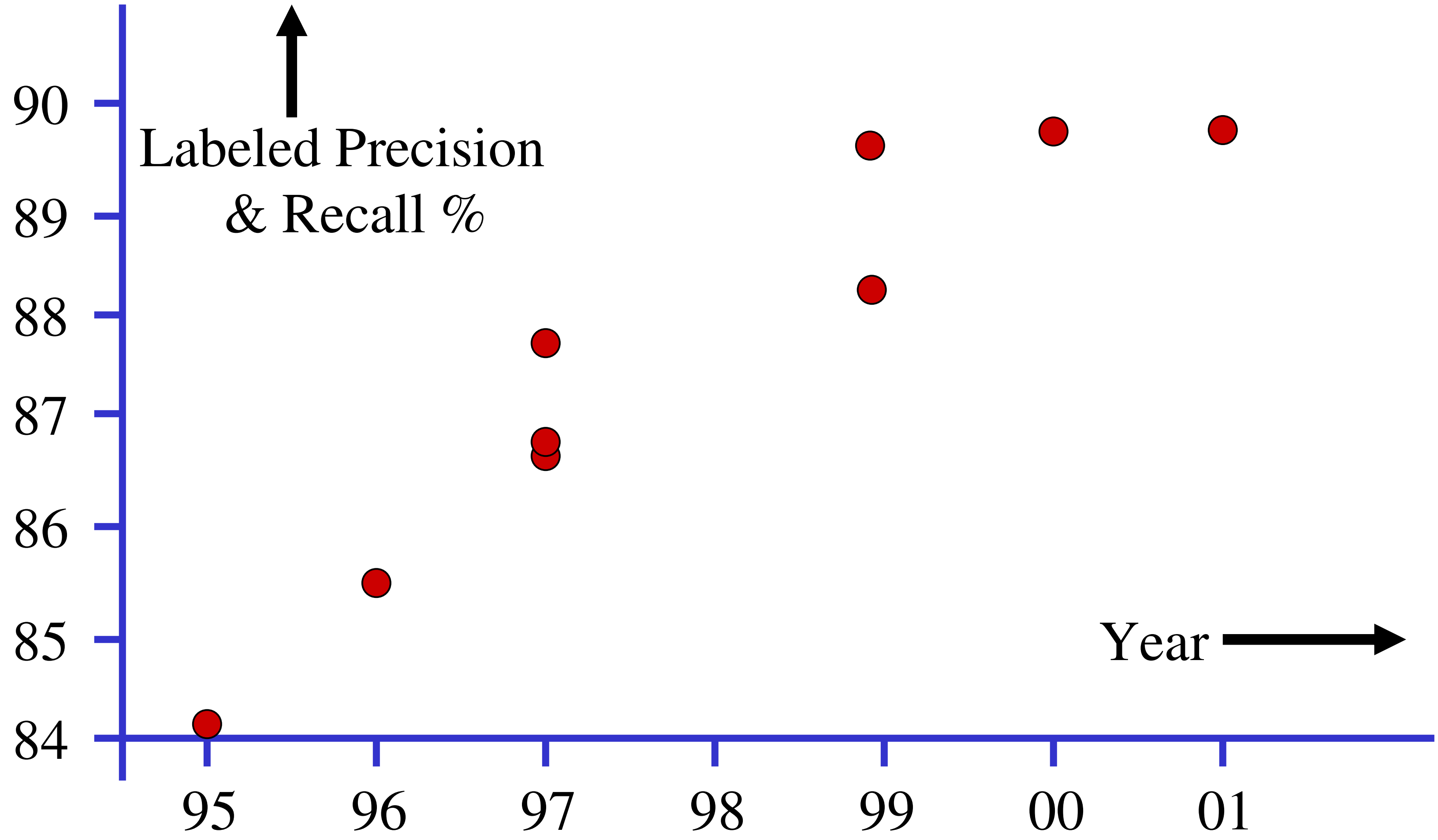
Lexicalized Probabilities for Heads

- p(prices | n-plural) = .013
- p(prices | n-plural, NP) = .013
- p(prices | n-plural, NP, S) = .025
- p(prices | n-plural, NP, S, v-past) = .052
- p(prices | n-plural, NP, S, v-past, fell) = .146

Statistical Parsing

- The last seven years have seen a rapid improvement of statistical parsers due to lexicalization.
- Contributors to this literature include Bod, Collins, Johnson, Magerman, Ratnaparkhi, and myself .

Statistical Parser Improvement



New Parsing Results

I Very Old Parsing Results (1990-2004)

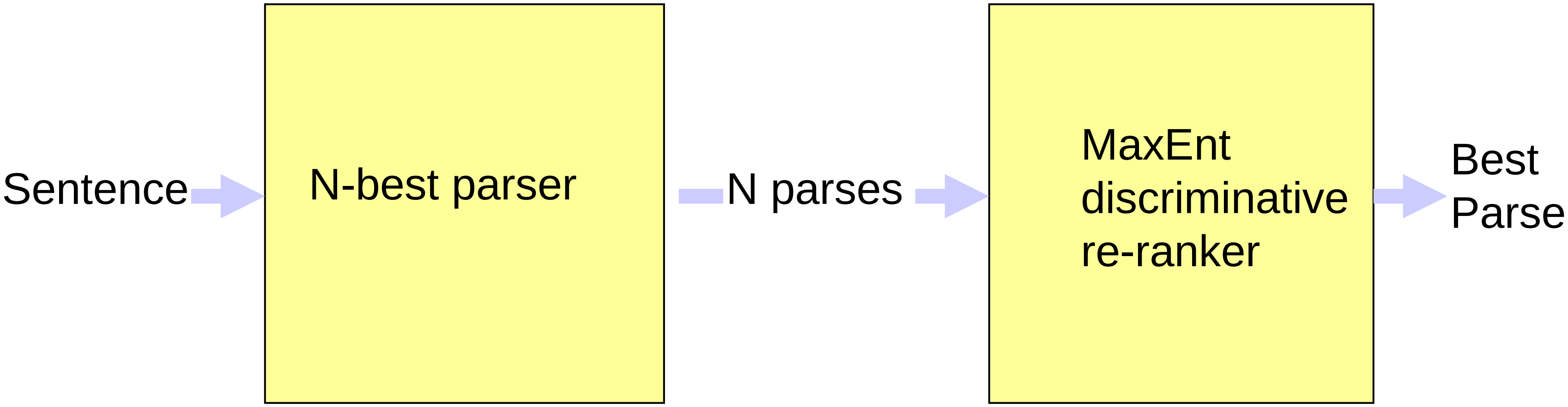
II More Recent Old Parsing Results (2004-2005)

III Learning from Unlabeled Data

IV Beyond the Wall-Street Journal

V Future Work

Coarse-to-fine n-best parsing and MaxEnt discriminative re-ranking


Sentence

N-best parser

N parses

MaxEnt discriminative re-ranker

Best Parse

50-best parsing results (oracle)

	1-best	2-best	10-best	25-best	50-best
F MEASURE	0.897	0.914	0.948	0.960	0.968

Base rate for the parser

MaxEnt Re-ranking

- N-best parses

$$Y(s) = \{y_1(s), y_2(s), \ldots, y_n(s)\}$$

- M re-ranking features

$$F = \{f_1, f_2, \ldots, f_m\}$$

- M weights, one for each feature

$$\Phi = \{\phi_1, \phi_2, \ldots, \phi_m\}$$

- Value of parse

$$V(y) = F(y) \bullet \Phi = \sum_{i=1}^{m} f_i(y)\phi_i$$

- Find highest-valued parse

$$\hat{y}(s) = \arg\max_{y \in Y(s)} V(y)$$

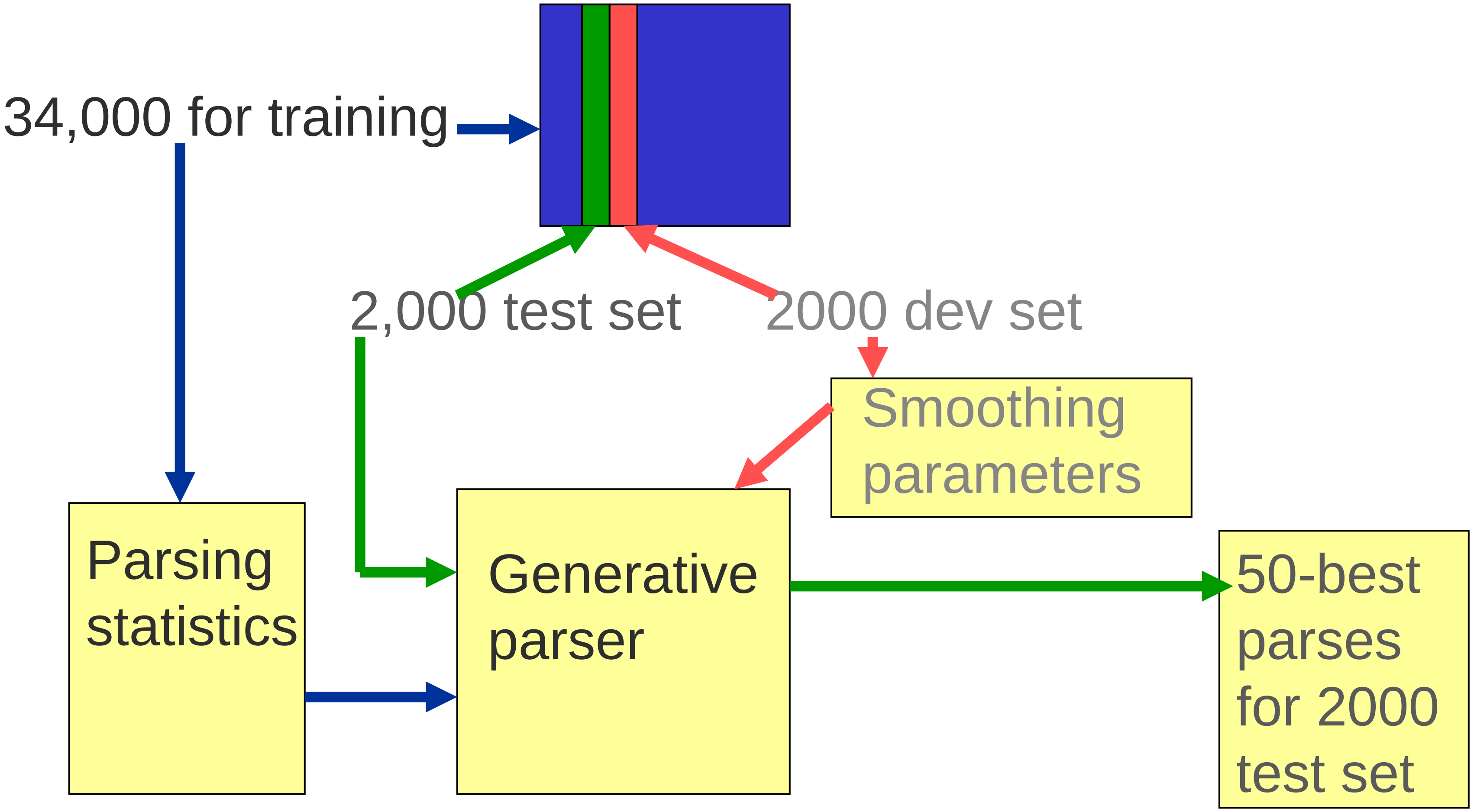
Feature Schema

- There are 13 feature schema, for example
- Number of constituents of { length k X constituents are at end of sentence, before a punctuation mark, neither} There are 1049 different versions of this feature schema, e.g., the parse has 2-4 constituents of length 5-8 at end of sentence.

Features Continued

- Only schema instantiations with five or more occurrences are retained.
- There are 1,141,697 features.
- For any given parse, almost all will have value zero.
- There are functions which map from parses to features with non-zero values to make the computation efficient.

Cross validation for feature-weight estimation

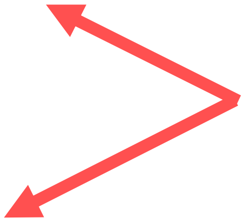


Collins (2000)

Discriminative Parsing Results

- "Standard" setup. Trained on sections 2-22, 24 for development, tested on section 23, sentences of length ≤ 100

	F-measure
New Parser	0.913
Re-ranked Collins n-best	0.904
Charniak 2000	0.895



17% error reduction

New Parsing Results

I Very Old Parsing Results (1990-2004)

II More Recent Old Parsing Results (2004-2005)

III Learning from Unlabeled Data

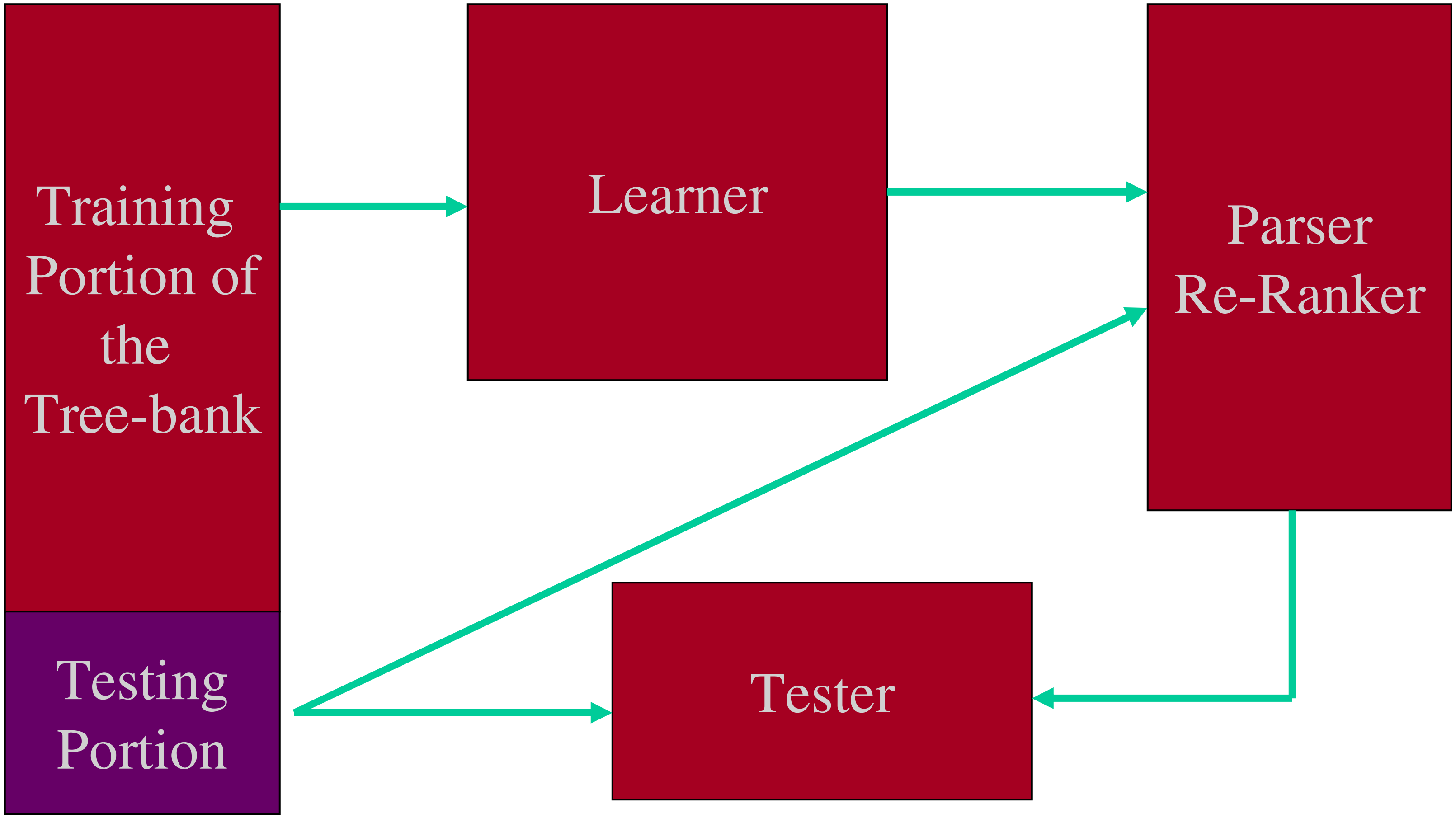
IV Beyond the Wall-Street Journal

V Future Work

Self Training

- Previously we trained on the human parsed Penn Tree-bank.
- It would be beneficial if we could use more, unparsed, data to “learn” to parse better.
- Self training is using the output of a system as extra training data for the system.

The Basic Paradigm



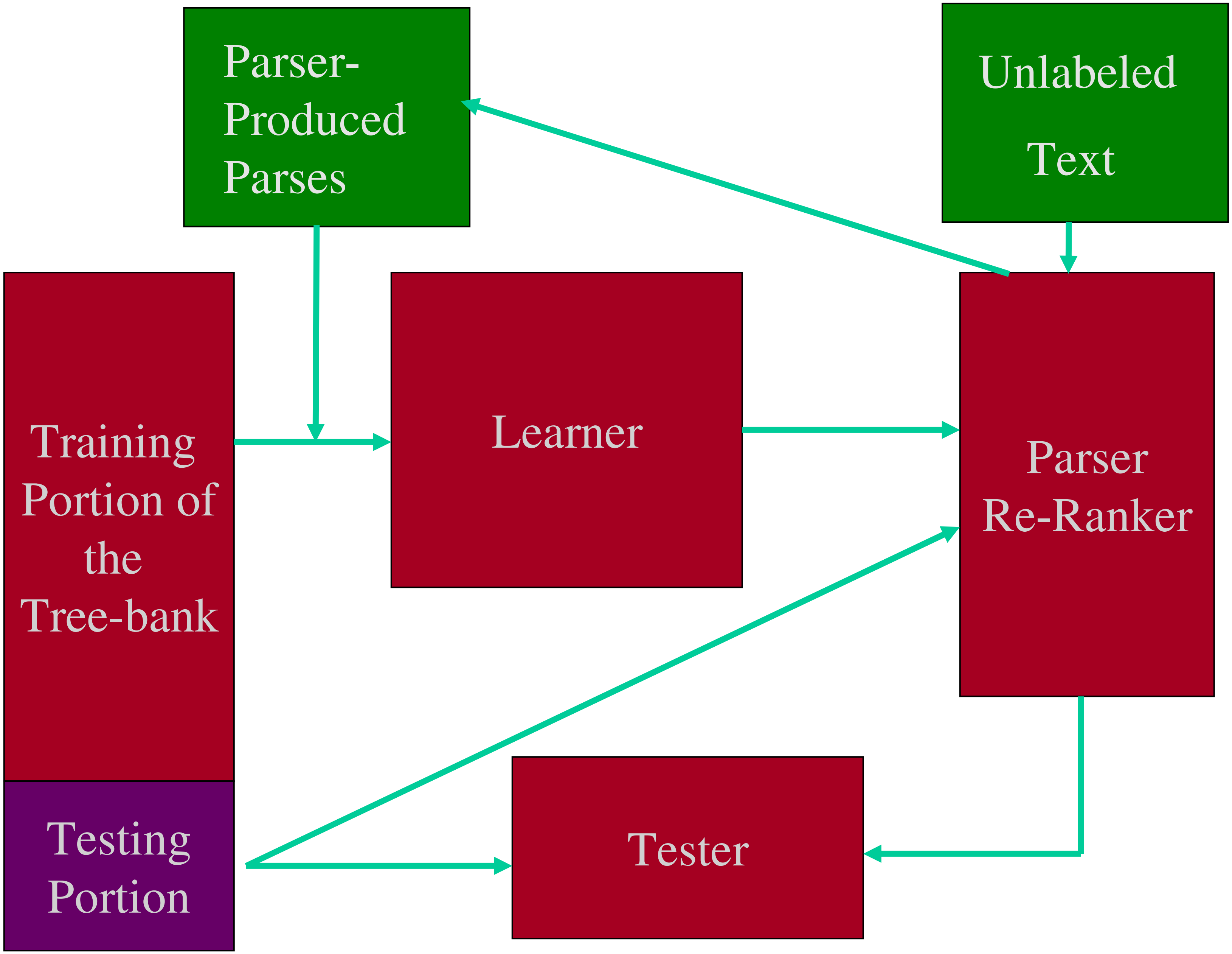
Training Portion of the Tree-bank

Testing Portion

Learner

Parser Re-Ranker

Tester



Parser-
Produced
Parses
Unlabeled
Text
Training
Portion of
the
Tree-bank
Learner
Parser
Re-Ranker
Testing
Portion
Tester

Self Training

- The trouble is, it is “known” not to work.

“Nor does self-training … offer a way out. Charniak(1997) showed a small improvement from training on 40M words of its own output, probably because the increased size of the training set gave somewhat better counts for the head dependency model. However, training on data that merely confirms the parser’s existing view cannot help in any other way and one might expect such improvements to be transient and eventually to degrade in the face of otherwise noisy data.” (Steedman et. Al. 2002)

Experimental Setup

- Parsed and re-ranked 400 million words of "North-American News" text.
- Added 20 million words from LA Times to training data.
- Parsed Section 23 of Penn Tree-bank as before

Self-Training Results

- Charniak 2000 parser 89.5%
- With re-ranking 91.3%
- With extra 40 million words 92.1

25% error reduction

What is Going On?

- Without the re-ranker self-training does not seem to work.
- With the re-ranker, it does.

- We have no idea why.

New Parsing Results

I Very Old Parsing Results (1990-2004)

II More Recent Old Parsing Results (2004-2005)

III Learning from Unlabeled Data

IV Beyond the Wall-Street Journal

V Future Work

How does the Parser Work on Corpora other than WSJ?

- Very badly. 83.9% on the Brown Corpus
- The Brown Corpus is a "balanced" corpus of 1,000,000 words of text from 1961: Newspapers, novels, religion, etc.
- Are our parsers over-bread racehorses, finely tuned to Wall-Street Journal?

Results on the Brown Corpus

• Basic parser on WSJ	89.5%	
• Basic parser on Brown	83.9%	←
• Re-ranking parser on Brown	85.8%	
• Re-ranking + 40million words	87.7%	←

83.9 to 87.7 is a 24% error reduction

New Parsing Results

I Very Old Parsing Results (1990-2004)

II More Recent Old Parsing Results (2004-2005)

III Learning from Unlabeled Data

IV Beyond the Wall-Street Journal

V Future Work

What Next?

- More experiments on self-training and new domains. E.g., would self-training on a biology textbook help parse medical text?
- We have some really way-out ideas on repeated parsing the same text using different grammars. A comparatively tame version of this has been written up for NAACL.

Conclusion

- We have improved parsing a lot in the last year.

- We have lots of wacky ideas for the future.

Thank You

Why do we Need to Use Tree-bank Non-terminals?

- Because our output will be marked incorrect otherwise.
- So we only need them at output, before then we can use whatever ones we want.
- Why might we want other ones?

We are Already Using Different Non-Terminals

- When we condition the choice of a word on another word we are in-effect creating new non-terminals

- P(prices | NNS, subject of “fell”) = 0.25
- P(NP-NNS^S-fell → prices) = 0.25

This is just a funny non-terminal

Clustering Non-Terminals

- So let us create lots of really specific non-terminals and cluster them, e.g.,
- NonTrm1005 = "A noun phrase headed by "prices" under a sentence headed by "fell"
- ClusterNonTerm223

  = {NonTrm1005, NonTerm2200 …}

  ClusterNonTerm223 → ClusterNonTerm111

Problem: Optimal Clustering is NP-Hard

- Solution 1: Don’t worry, many cluster algorithms do quite well anyway.
- Solution 2: Do many different clusterings.