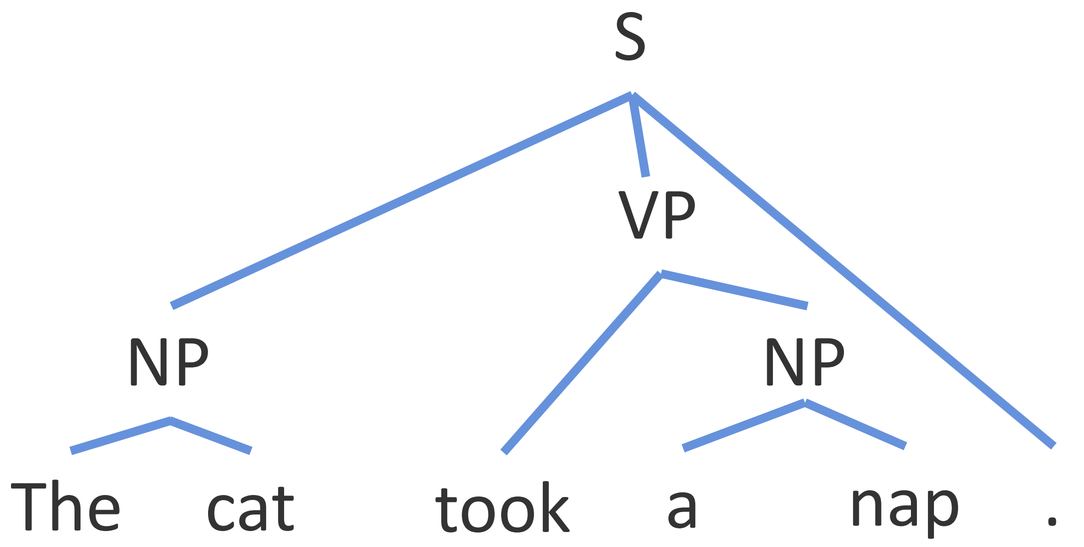
Policy Gradient as a Proxy for Dynamic Oracles in Constituency Parsing

Berkeley
N L P

Daniel Fried and Dan Klein

Parsing by Local Decisions



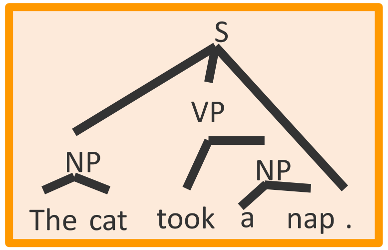
$\boxed{\text{(S}} \rightarrow \boxed{\text{(NP}} \rightarrow \boxed{\text{The}} \rightarrow \boxed{\text{cat}} \rightarrow \boxed{\text{)}} \rightarrow \boxed{\text{(VP}} \rightarrow \ldots$

$$L(\theta) = \log p(y|x;\theta) = \sum_t \log p(y_t | y_{1:t-1}, x; \theta)$$

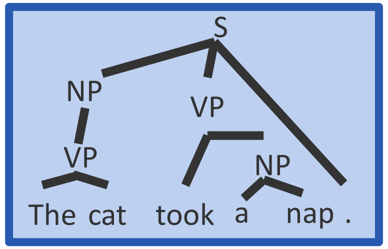
Non-local Consequences

Loss-Evaluation Mismatch

y



$\hat{y}$



$$\Delta(y, \hat{y})\text{: -F1}(y, \hat{y})$$

Exposure Bias

True Parse	y	(S →	(NP →	The →	cat →	…
Prediction	$\hat{y}$	(S →	(NP →	(VP →	??	

[Ranzato et al. 2016; Wiseman and Rush 2016]

Dynamic Oracle Training

Explore at training time. Supervise each state with an expert policy.

True Parse y: (S → (NP → The → cat → ...

addresses exposure bias

Prediction (sample, or greedy) $\hat{y}$: (S → (NP → (VP → The → ...

Oracle y^*: (NP, The, The, cat

$$L(\theta) = \sum_t \log p(y_t^* | \hat{y}_{1:t-1}, x; \theta)$$

choose y_t^* to maximize achievable F1 (typically)

addresses loss mismatch

[Goldberg & Nivre 2012; Ballesteros et al. 2016; inter alia]

Dynamic Oracles Help!

Expert Policies / Dynamic Oracles

Daume III et al., 2009; Ross et al., 2011;
Choi and Palmer, 2011; Goldberg and Nivre, 2012;
Chang et al., 2015; Ballesteros et al., 2016; Stern et al. 2017

mostly dependency parsing

PTB Constituency Parsing F1

System	Static Oracle	Dynamic Oracle
Coavoux and Crabbé, 2016	88.6	89.0
Cross and Huang, 2016	91.0	91.3
Fernández-González and Gómez-Rodríguez, 2018	91.5	91.7

What if we don’t have a dynamic oracle?

Use reinforcement learning

Reinforcement Learning Helps! (in other tasks)

machine translation

Auli and Gao, 2014; Ranzato et al., 2016; Shen et al., 2016

Xu et al., 2016; Wiseman and Rush, 2016; Edunov et al. 2017

CCG parsing

several, including dependency parsing

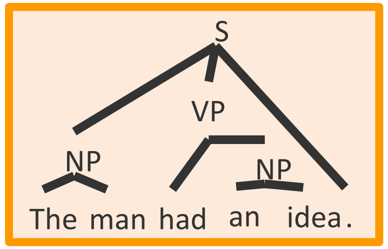
machine translation

Policy Gradient Training

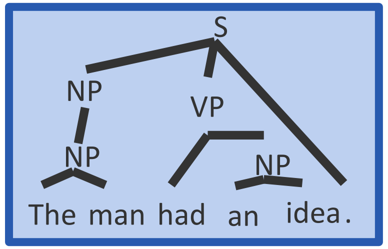
Minimize expected sequence-level cost:

$$R(\theta) = \sum_{\hat{y}} p(\hat{y}|x;\theta)\, \Delta(y,\hat{y})$$

True Parse y



Prediction $\hat{y}$



$\Delta(y,\hat{y})$

$$\nabla R(\theta) = \underbrace{\sum_{\hat{y}} p(\hat{y}|x;\theta)}_{\substack{\text{addresses}\\ \text{exposure bias}\\ \text{(compute by}\\ \text{sampling)}}}\, \underbrace{\Delta(y,\hat{y})}_{\substack{\text{addresses}\\ \text{loss}\\ \text{mismatch}\\ \text{(compute F1)}}}\, \underbrace{\nabla \log p(\hat{y}|x;\theta)}_{\substack{\text{compute in}\\ \text{the same way}\\ \text{as for the}\\ \text{true tree}}}$$

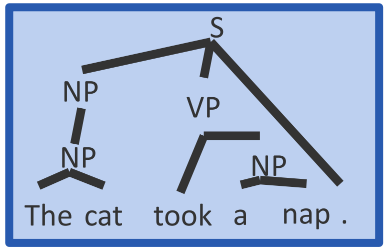
[Williams, 1992]

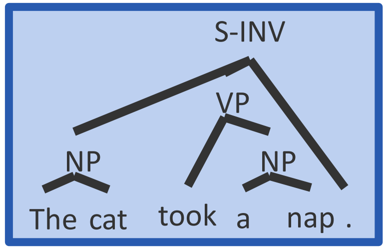
Policy Gradient Training

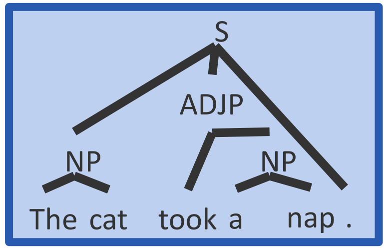
$$\nabla R(\theta) = \sum_{\hat{y}} p(\hat{y}|x;\theta)\ \Delta(y,\hat{y})\ \nabla \log p(\hat{y}|x;\theta)$$

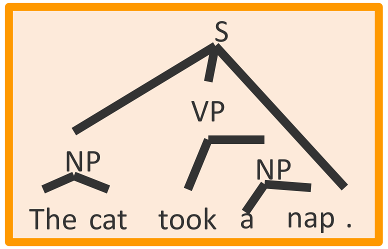
Input, x — The cat took a nap.

k candidates, $\hat{y}$









$\Delta(y,\hat{y})$ (negative F1)	-89	-80	-80	-100
	$*$	$*$	$*$	$*$
gradient for candidate	$\nabla \log p(\hat{y}_1|x;\theta)$	$\nabla \log p(\hat{y}_2|x;\theta)$	$\nabla \log p(\hat{y}_3|x;\theta)$	$\nabla \log p(y|x;\theta)$

Experiments

Setup

Parsers

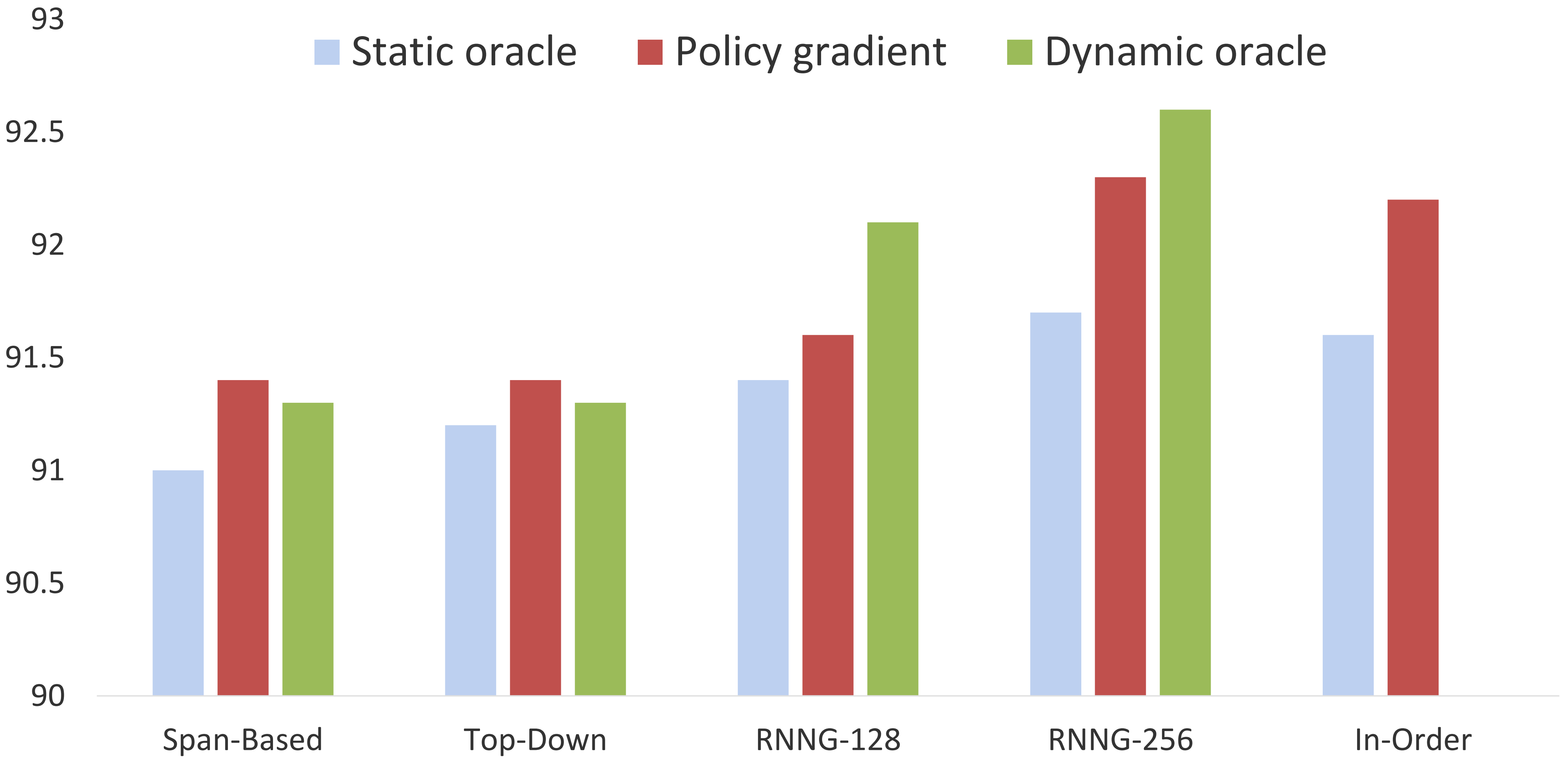
Span-Based [Cross & Huang, 2016]
Top-Down [Stern et al. 2016]
RNNG [Dyer et al. 2016]
In-Order [Liu and Zhang, 2017]

X

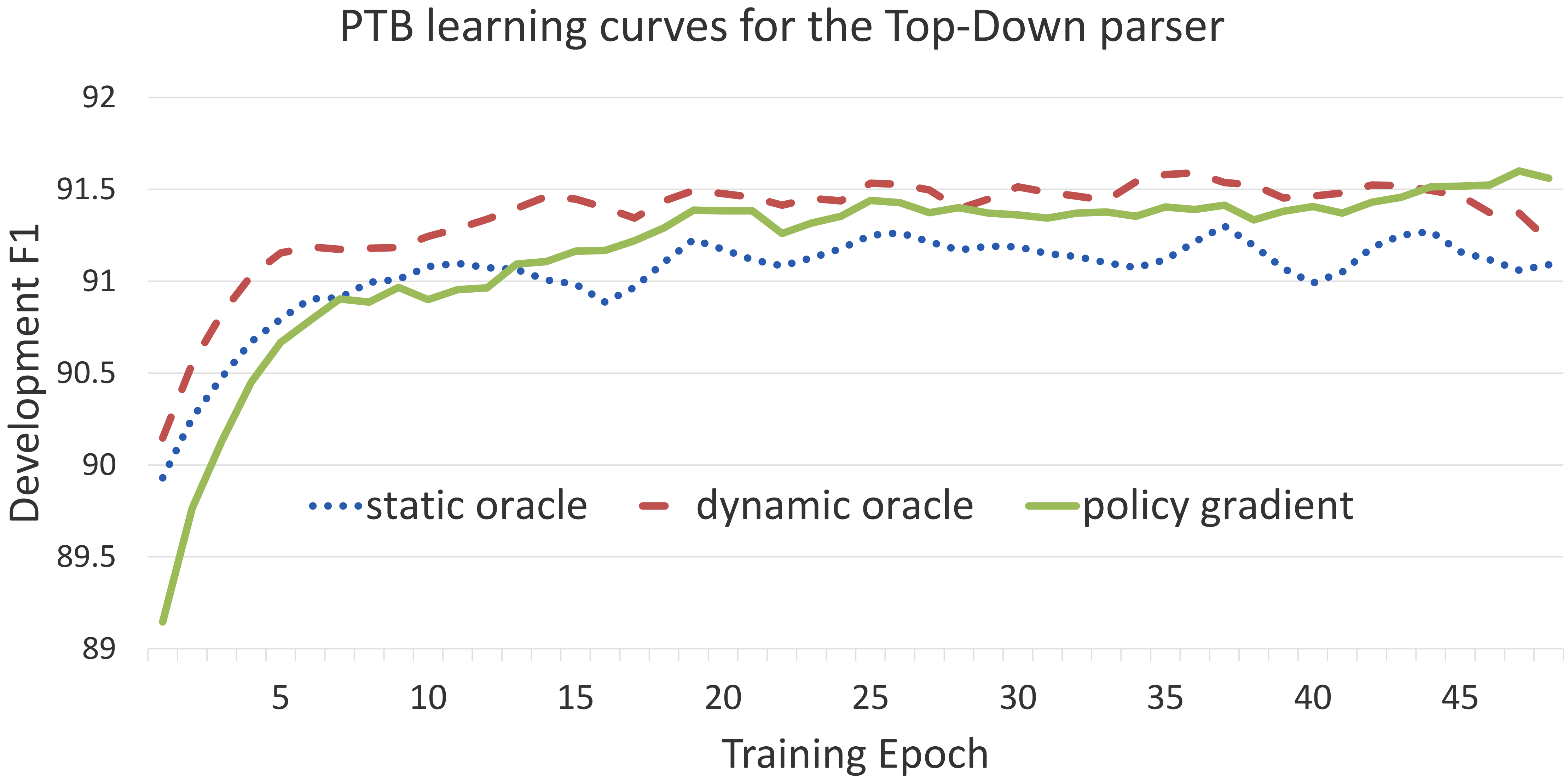
Training

Static oracle
Dynamic oracle
Policy gradient

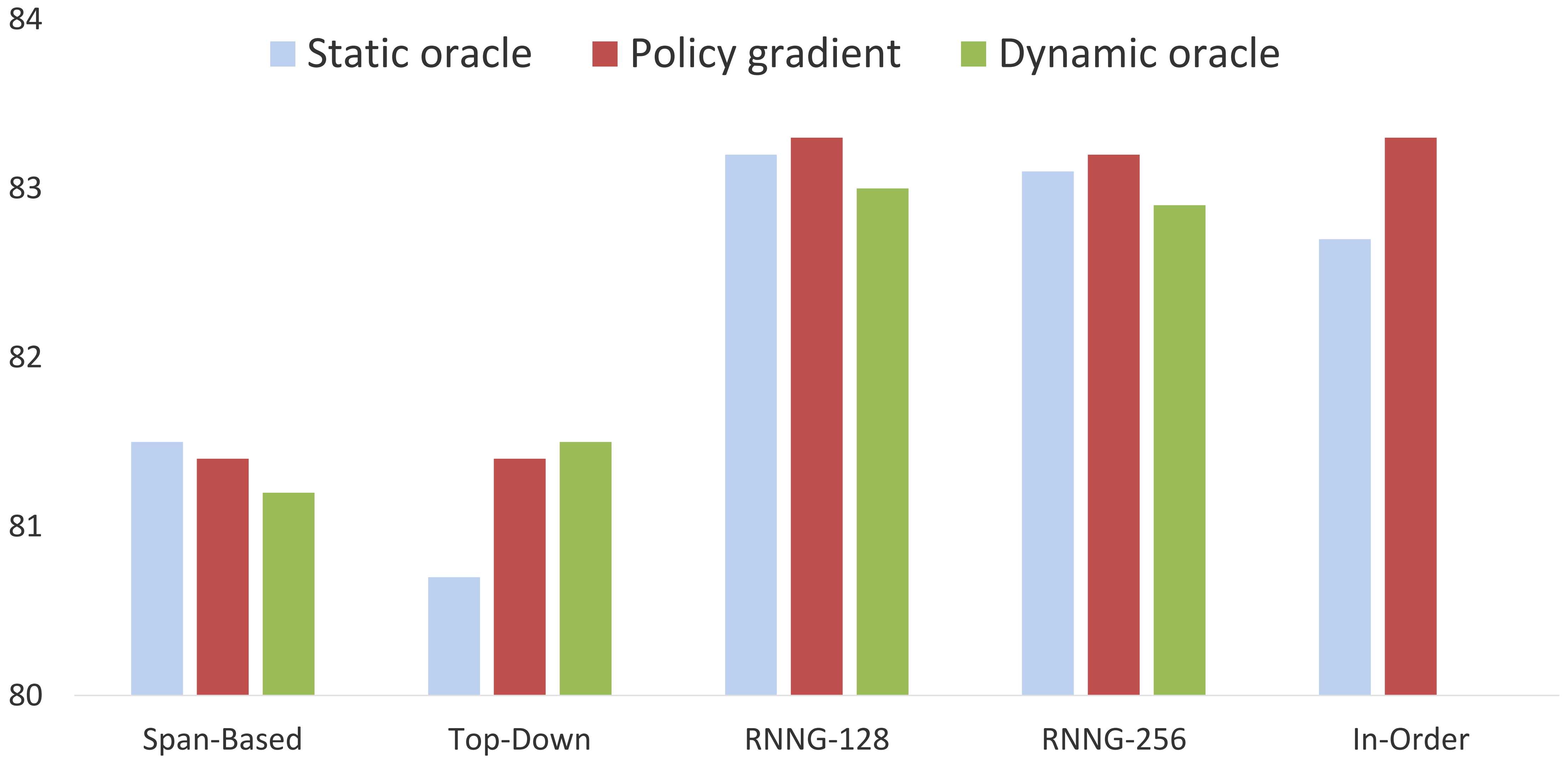
English PTB F1



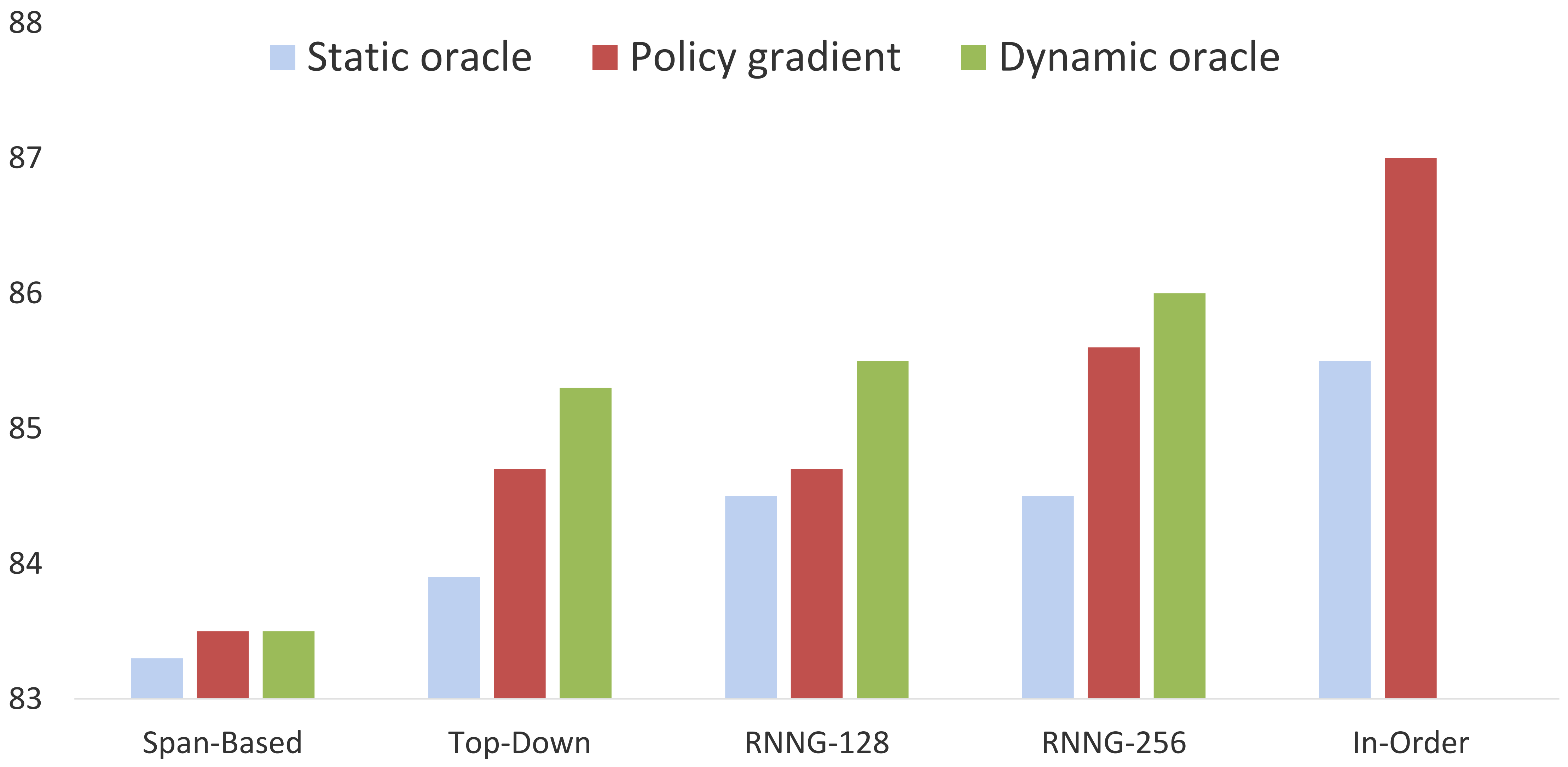
Training Efficiency



French Treebank F1



Chinese Penn Treebank v5.1 F1



Conclusions

- Local decisions can have non-local consequences
  - Loss mismatch
  - Exposure bias
- How to deal with the issues caused by local decisions?
  - Dynamic oracles: efficient, model specific
  - Policy gradient: slower to train, but general purpose

Thank you!

For Comparison: A Novel Oracle for RNNG

(S (NP The man) (VP had …

1. Close current constituent if it's a true constituent…

(S (NP The man)

… or it could never be a true constituent.

(S (VP (NP The man))

2. Otherwise, open the outermost unopened true constituent at this position.

(S (NP The man) (VP

3. Otherwise, shift the next word.

(S (NP The man) (VP had