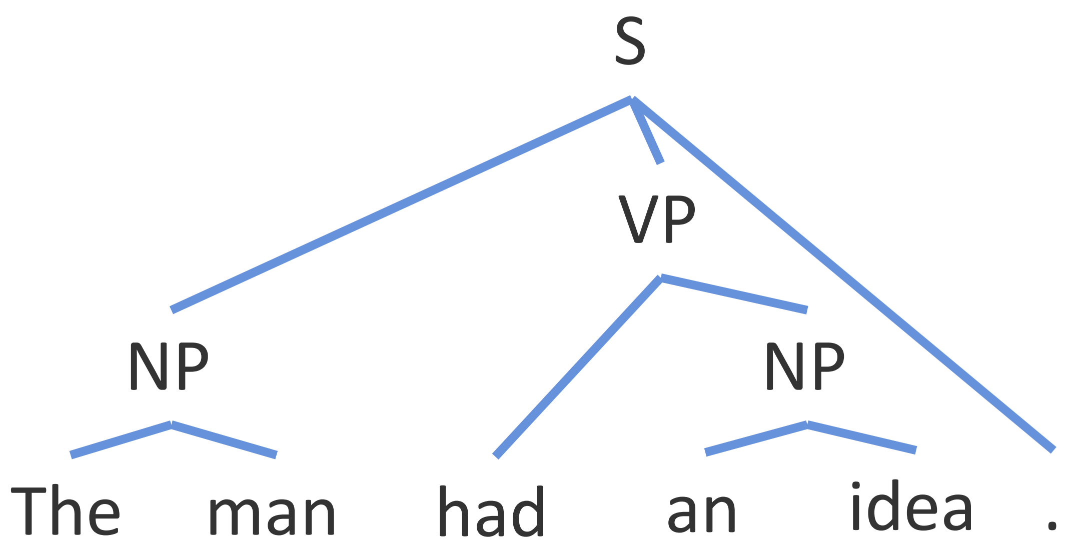
Improving Neural Parsing by Disentangling Model Combination and Reranking Effects

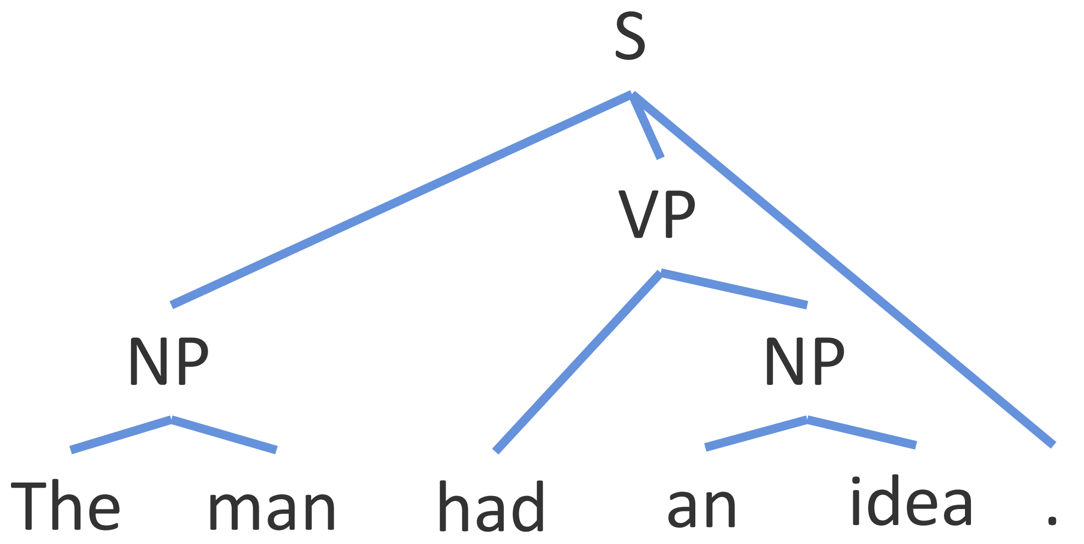
Daniel Fried*, Mitchell Stern* and Dan Klein
UC Berkeley

Top-down generative models

Top-down generative models

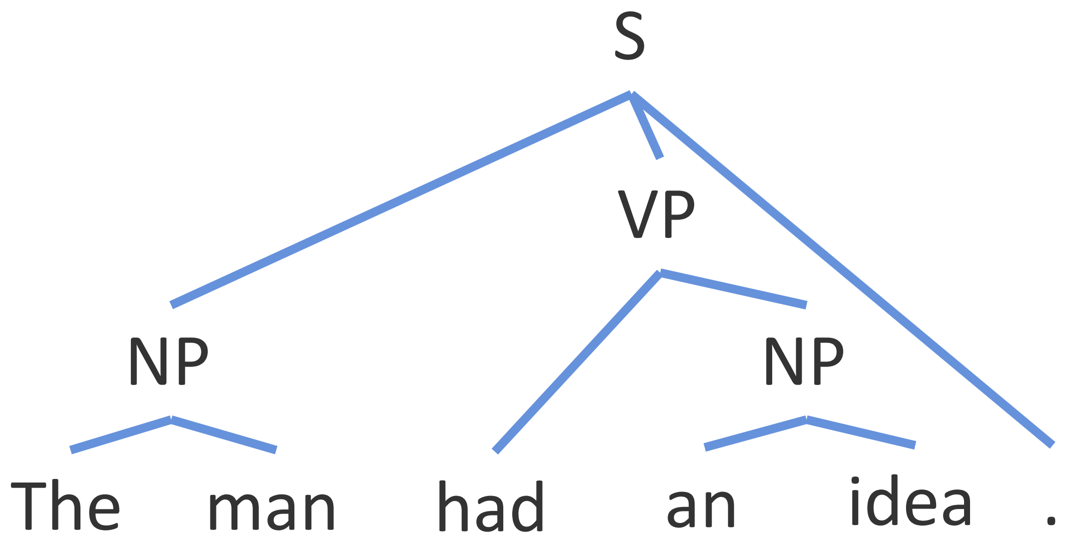


Top-down generative models



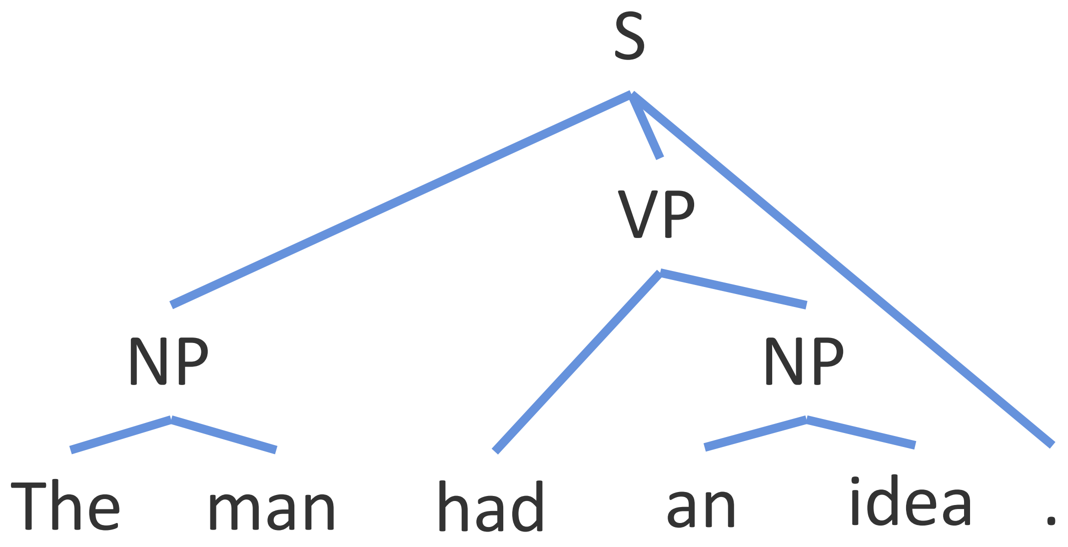
(S

Top-down generative models



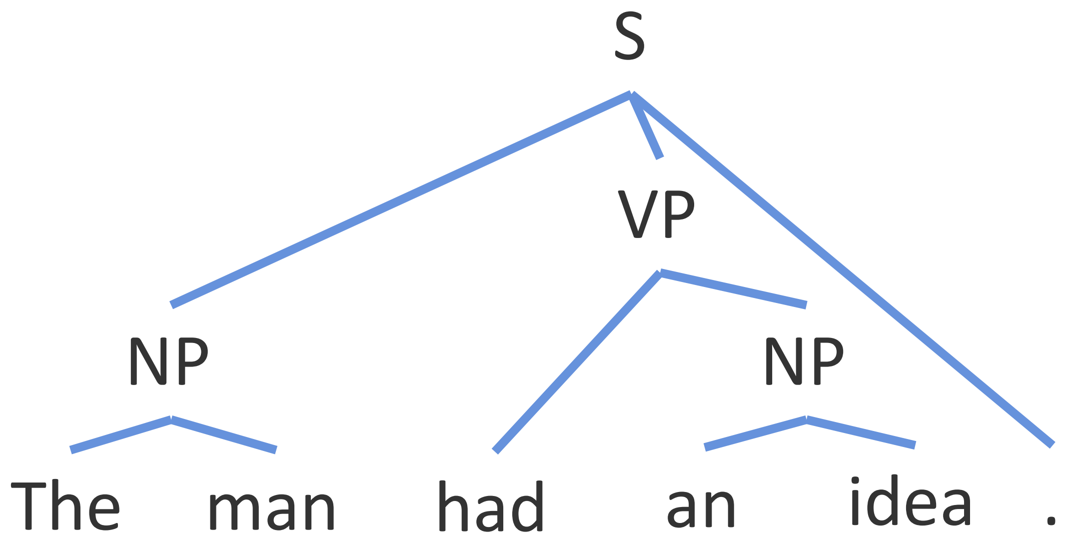
(S (NP

Top-down generative models



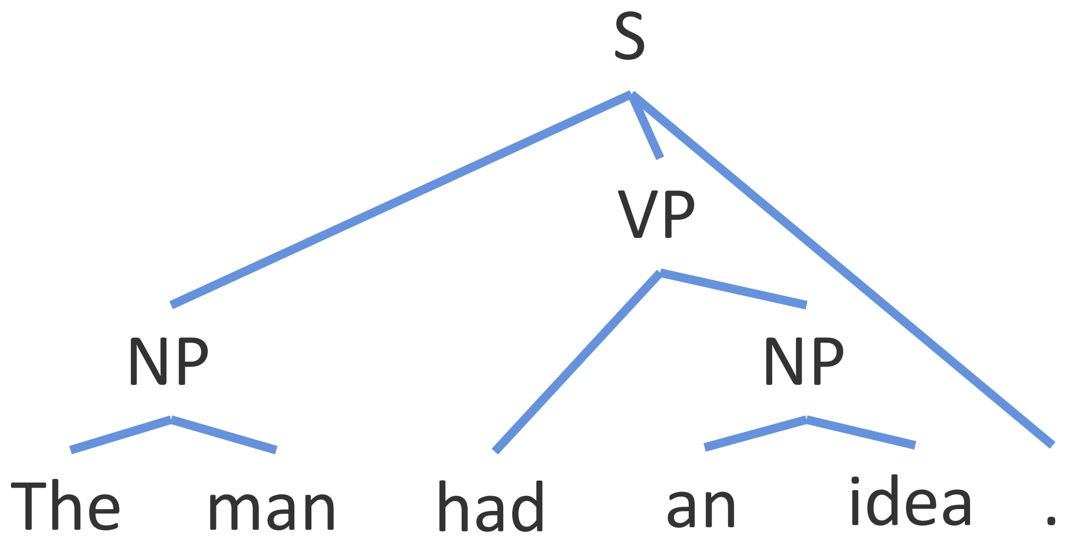
(S (NP The

Top-down generative models



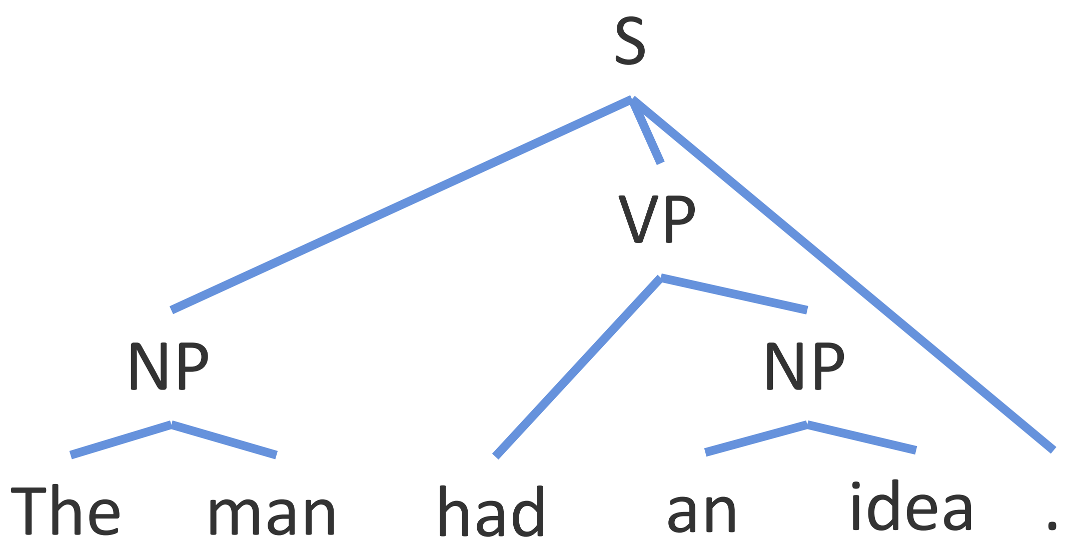
(S (NP The man

Top-down generative models



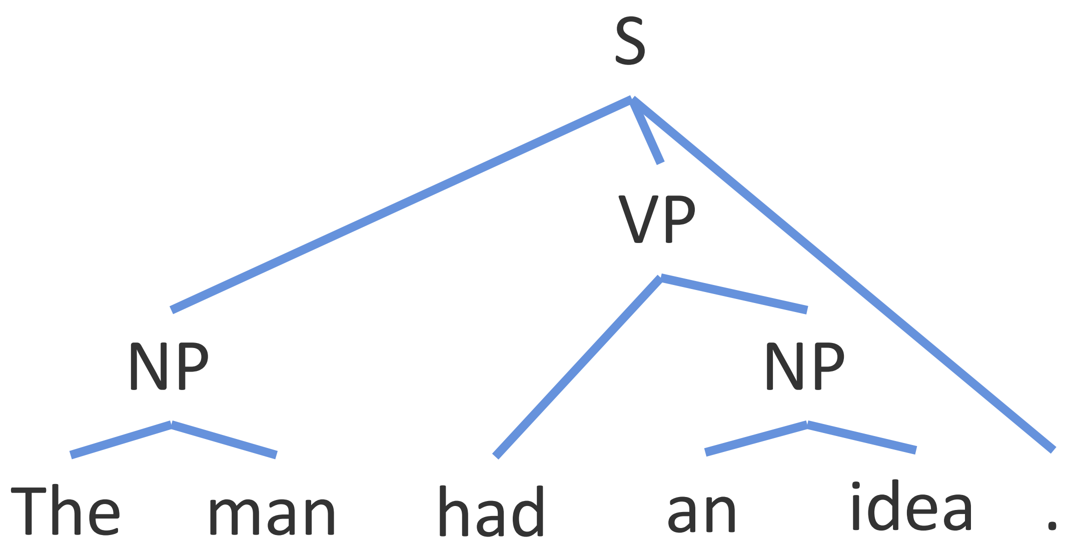
(S (NP The man)

Top-down generative models



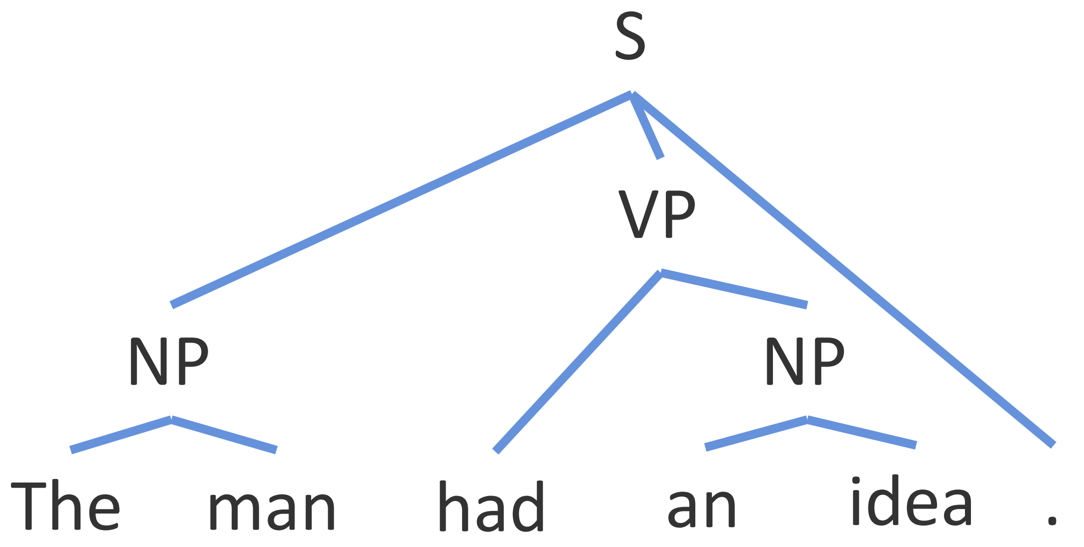
(S (NP The man) (VP

Top-down generative models



(S (NP The man) (VP had (NP an idea)) .)

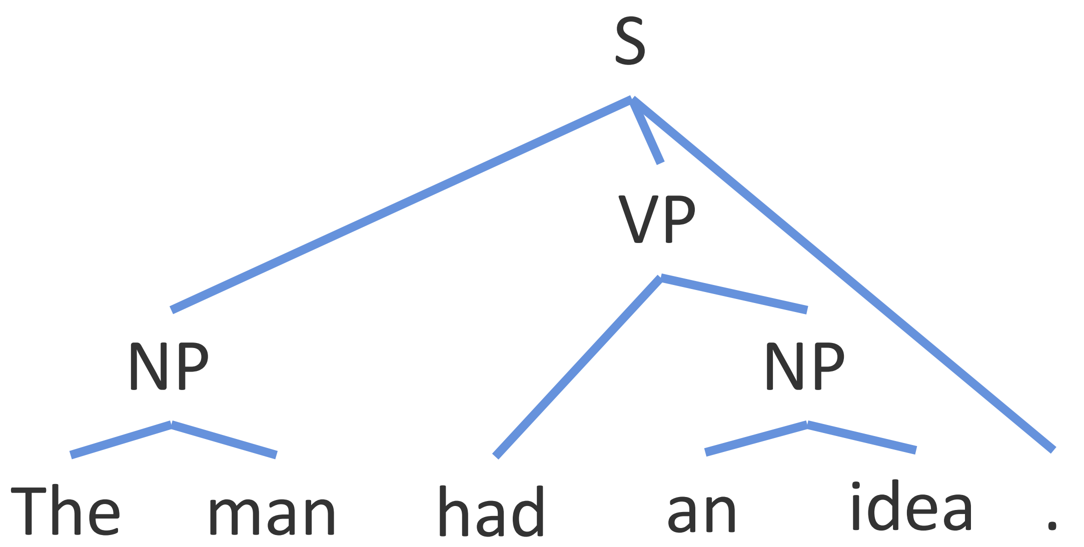
Top-down generative models



(S (NP The man) (VP had (NP an idea)) .)

$\mathbf{G}_{\text{LSTM}}$ [*Parsing as Language Modeling*, Choe and Charniak, 2016]

Top-down generative models



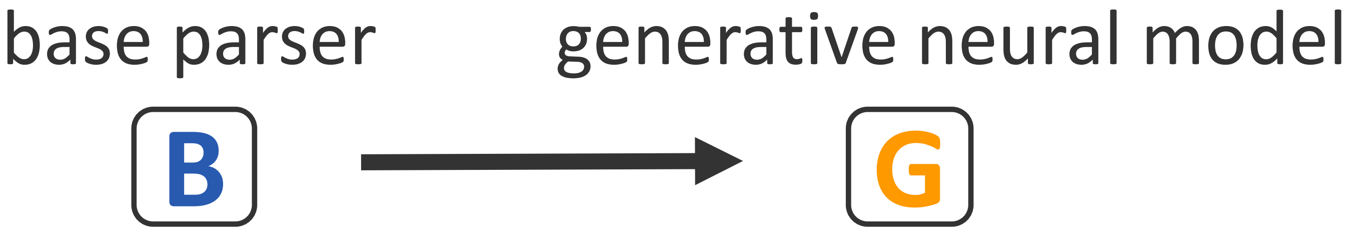
(S (NP The man) (VP had (NP an idea)) .)

G_{LSTM} [*Parsing as Language Modeling*, Choe and Charniak, 2016]

G_{RNNG} [*Recurrent Neural Network Grammars*, Dyer et al. 2016]

Generative models as rerankers

Generative models as rerankers



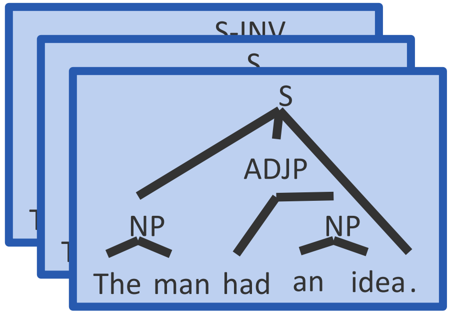
Generative models as rerankers

base parser

B

generative neural model

G



$$y \sim p_B(y|x)$$

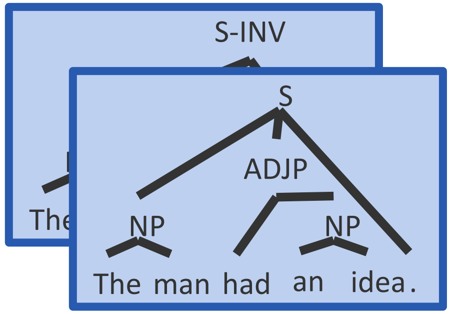
Generative models as rerankers

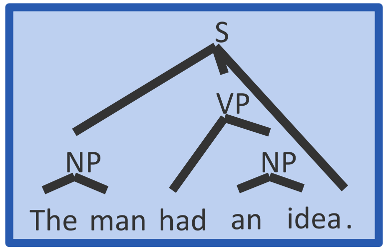
base parser

B

generative neural model

G

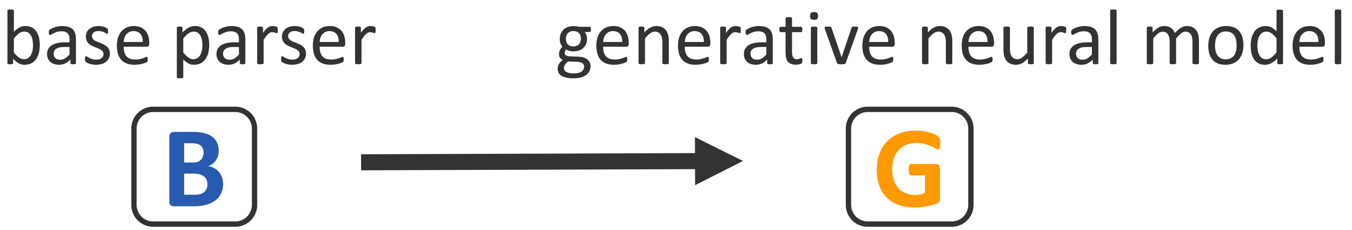




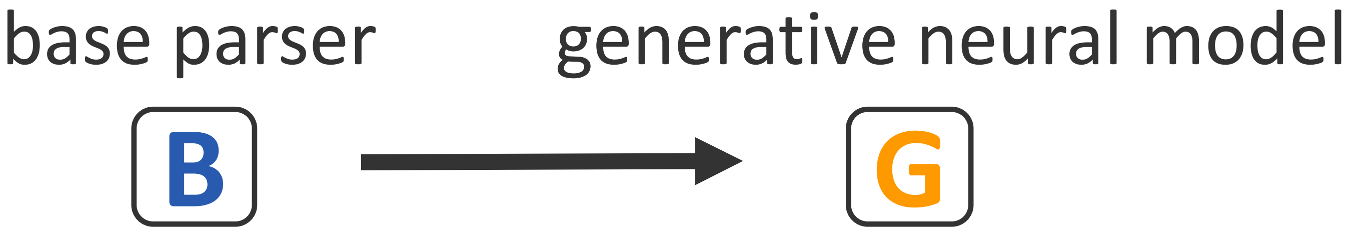
$y \sim p_{\boldsymbol{B}}(y|x)$

$\text{argmax}_y \; p_{\boldsymbol{G}}(x, y)$

Generative models as rerankers



Generative models as rerankers



F1 on Penn Tree Bank

Generative models as rerankers

base parser		generative neural model
B	→	**G**

F1 on Penn Tree Bank

Choe and Charniak 2016	**89.7** Charniak parser	→	**92.6** LSTM language model ($\mathbf{G}_{\mathrm{LSTM}}$)

Generative models as rerankers

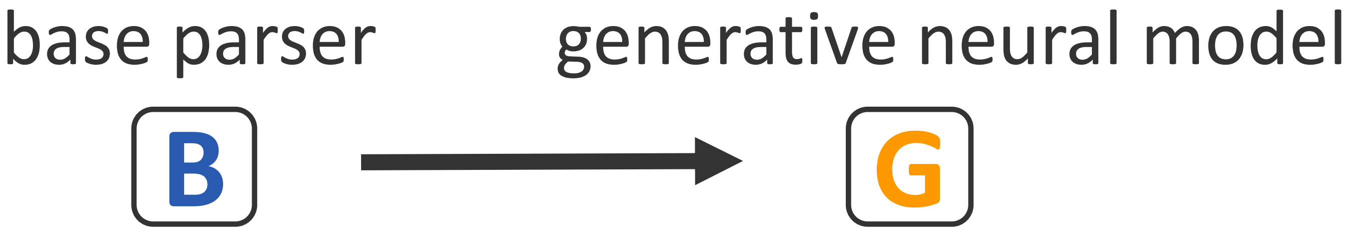
base parser → generative neural model

B → **G**

F1 on Penn Tree Bank

	Base parser		Generative model
Choe and Charniak 2016	**89.7** Charniak parser	→	**92.6** LSTM language model (**G**$_{LSTM}$)
Dyer et al. 2016	**91.7** RNNG-discriminative	→	**93.3** RNNG-generative (**G**$_{RNNG}$)

B: Necessary evil, or secret sauce?



B: Necessary evil, or secret sauce?

base parser — generative neural model

B → G

Should we try to do away with B?

B: Necessary evil, or secret sauce?

base parser　　generative neural model

B $\longrightarrow$ G

Should we try to do away with B?

No, better to combine B and G more explicitly

B: Necessary evil, or secret sauce?

base parser → generative neural model

$\boxed{B} \longrightarrow \boxed{G}$

Should we try to do away with B?

No, better to combine B and G more explicitly

93.9 F1 on PTB; **94.7** semi-supervised

Using standard beam search for G

True Parse	(S	(NP	The	man

Beam

Using standard beam search for G

True Parse (S (NP The man

(S

Beam

Using standard beam search for G

True Parse

(S (NP The man

Beam

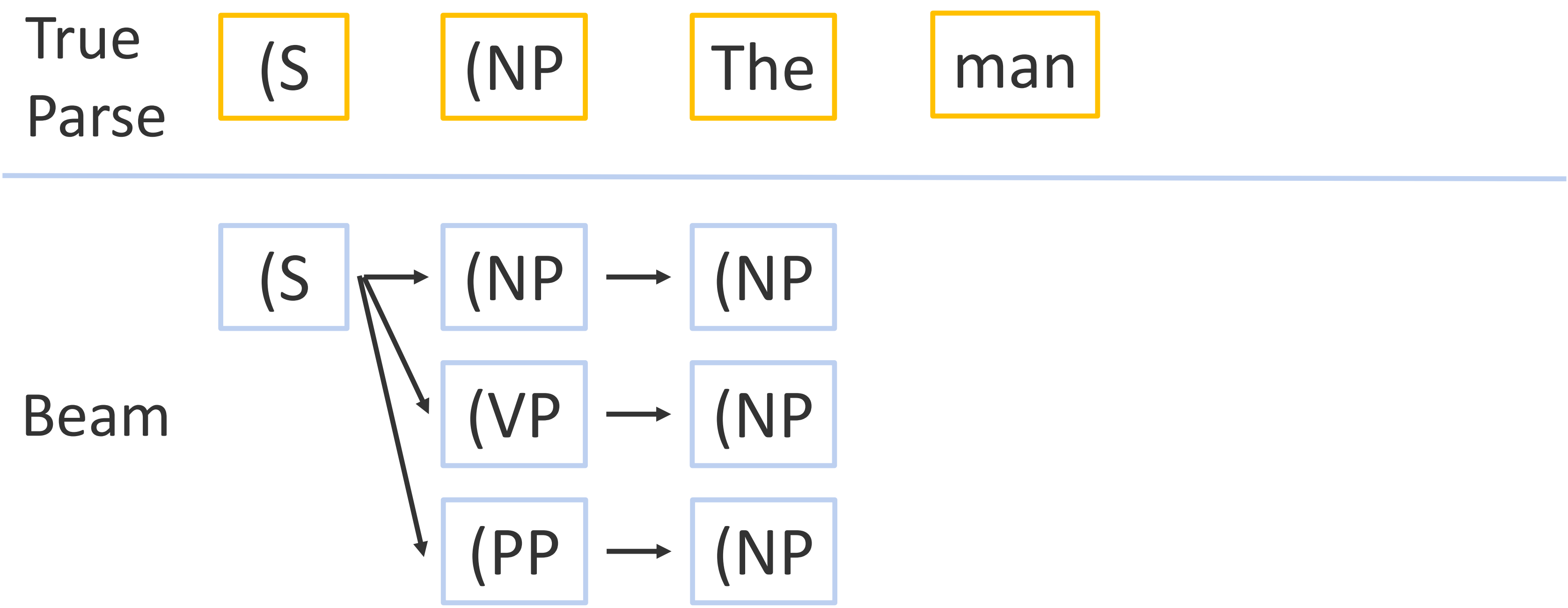
(S

(NP

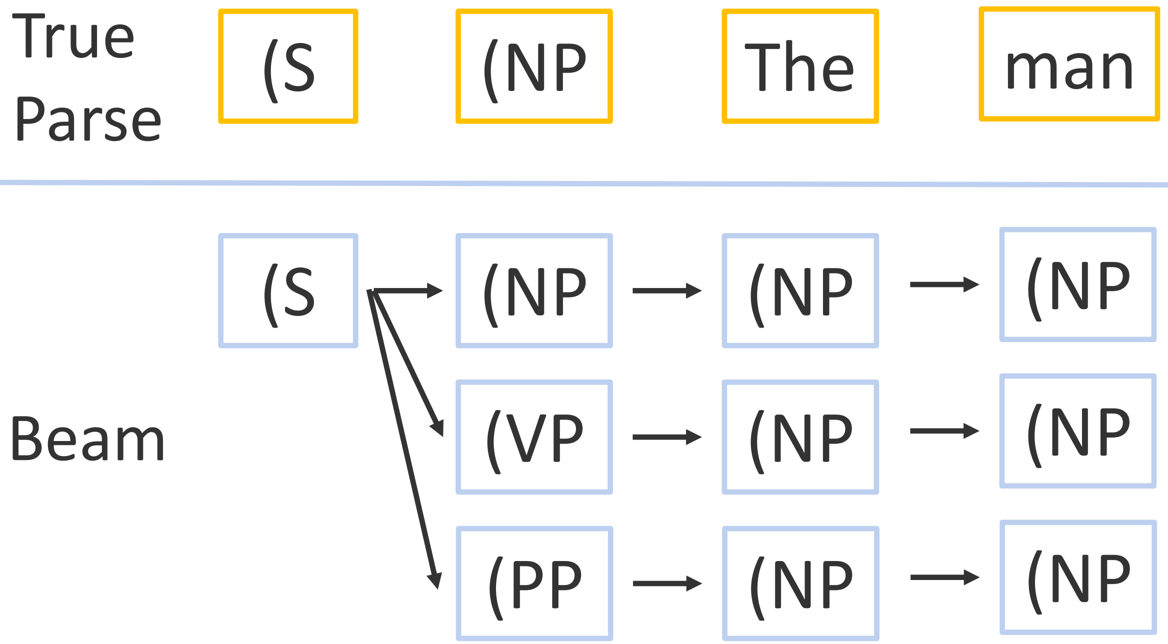
(VP

(PP

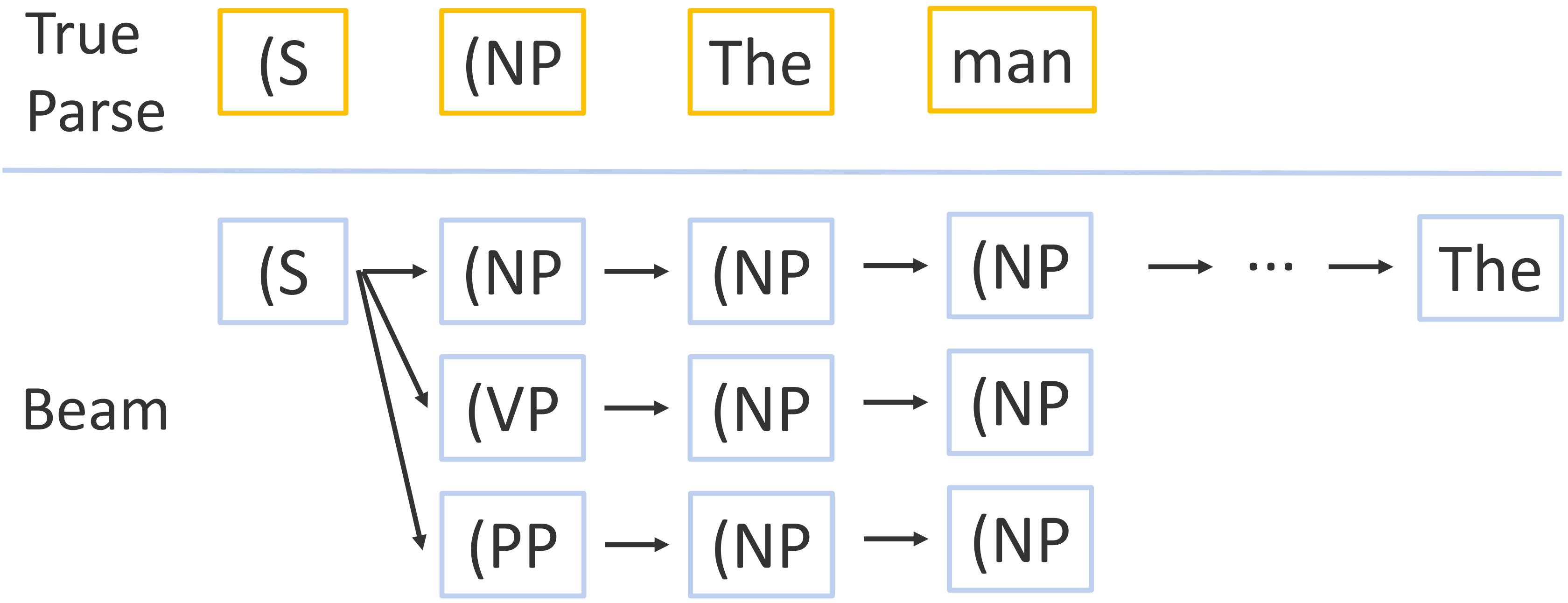
Using standard beam search for G



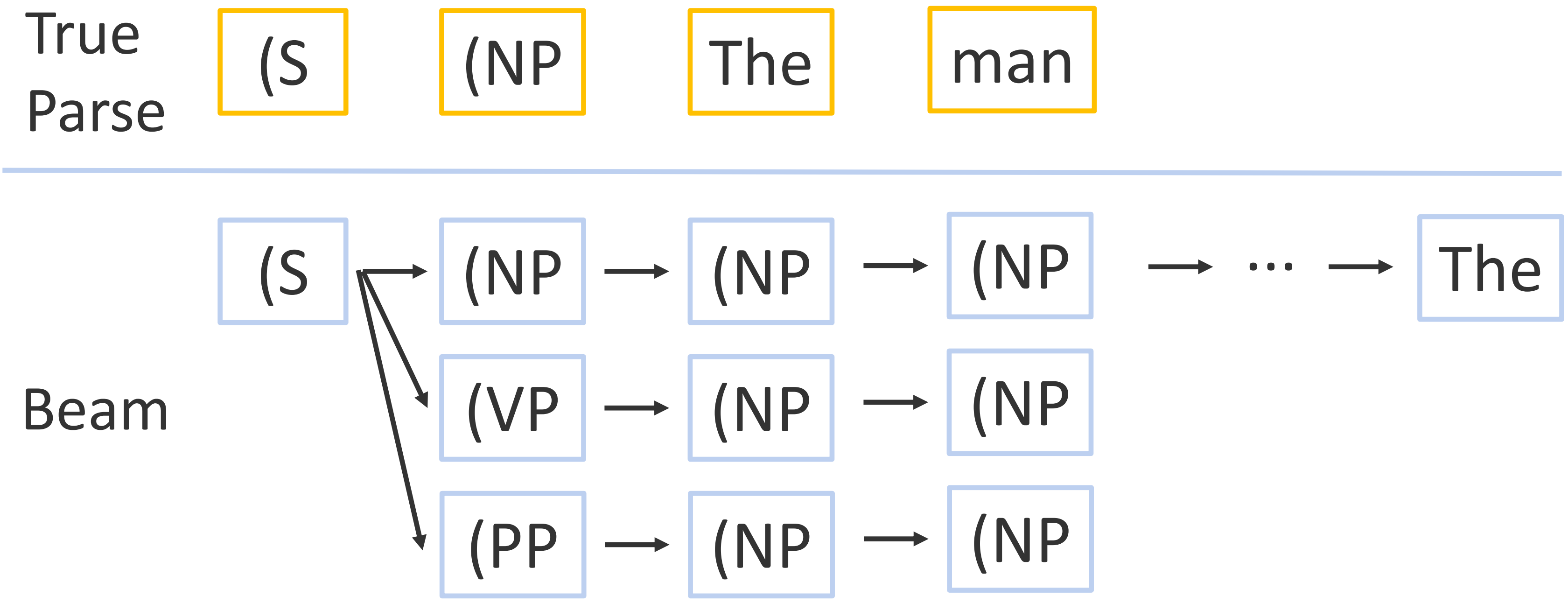
Using standard beam search for G



Using standard beam search for G



Using standard beam search for G



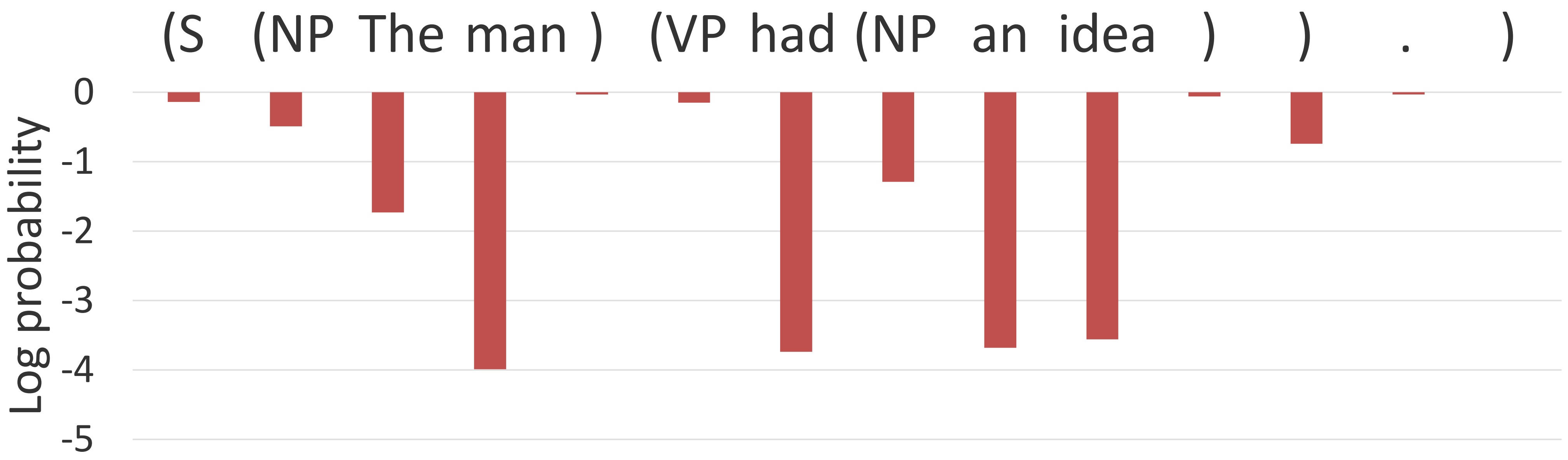
Beam Size 100

G_{RNNG} 29.1 F1

G_{LSTM} 27.4 F1

Standard beam search in G fails

Word generation is lexicalized:



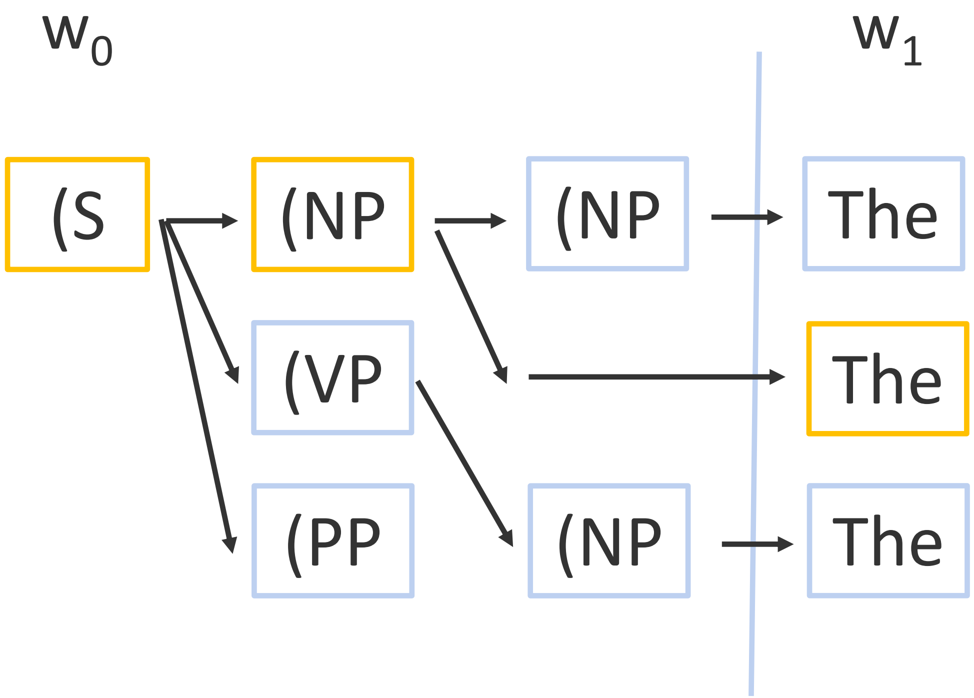
Word-synchronous beam search

w_0

$\boxed{\text{(S}}$

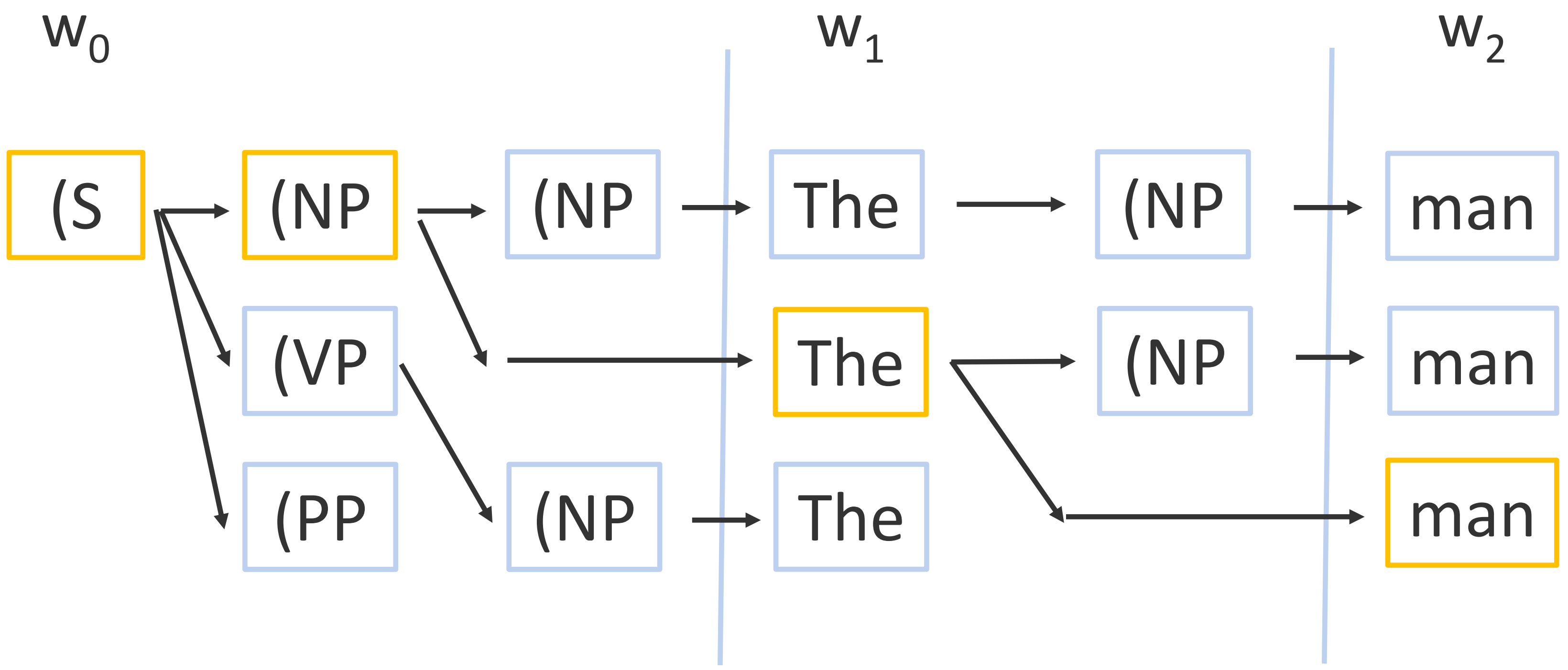
[Roark 2001; Titov and Henderson 2010; Charniak 2010; Buys and Blunsom 2015]

Word-synchronous beam search



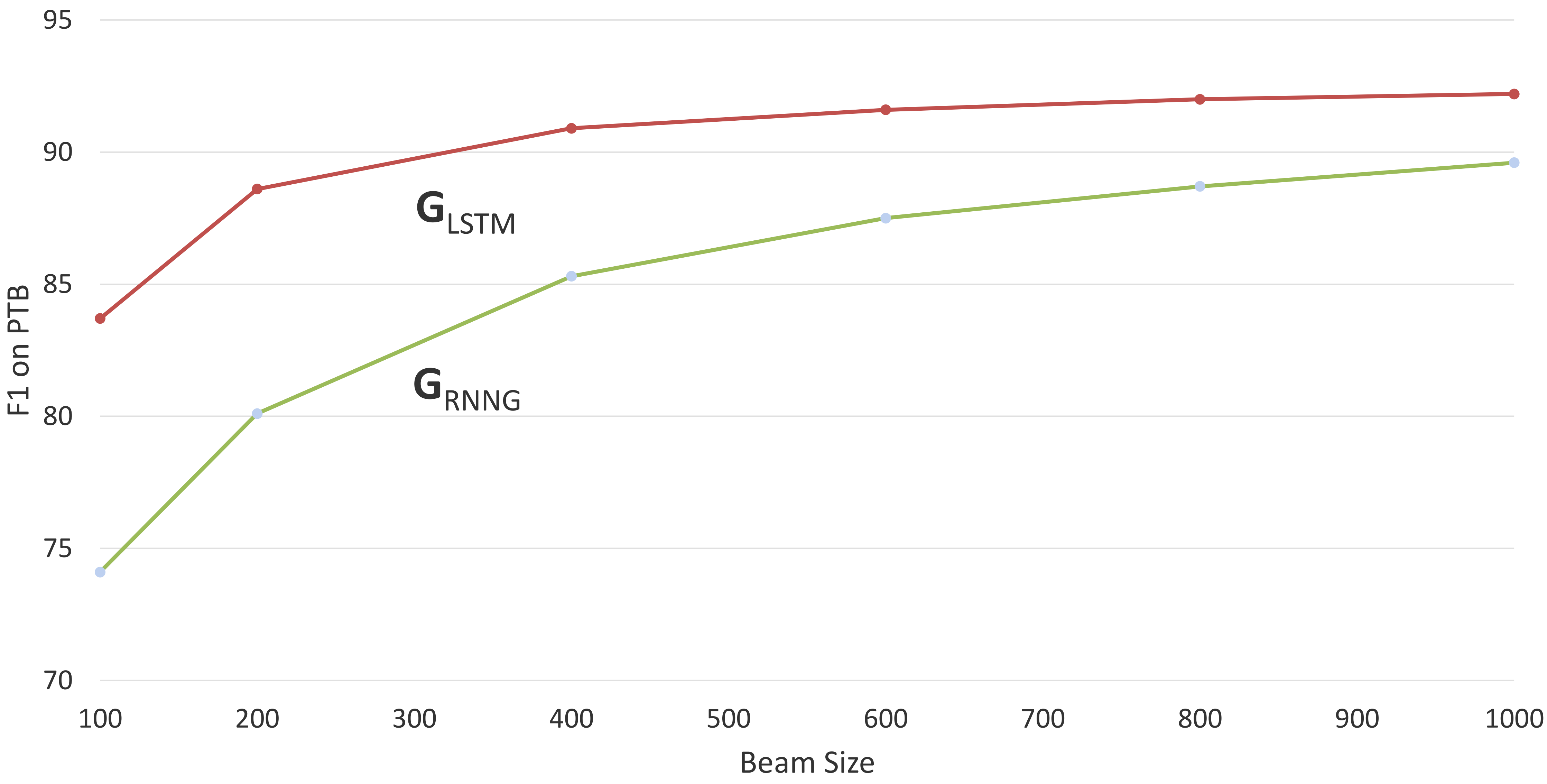
[Roark 2001; Titov and Henderson 2010; Charniak 2010; Buys and Blunsom 2015]

Word-synchronous beam search

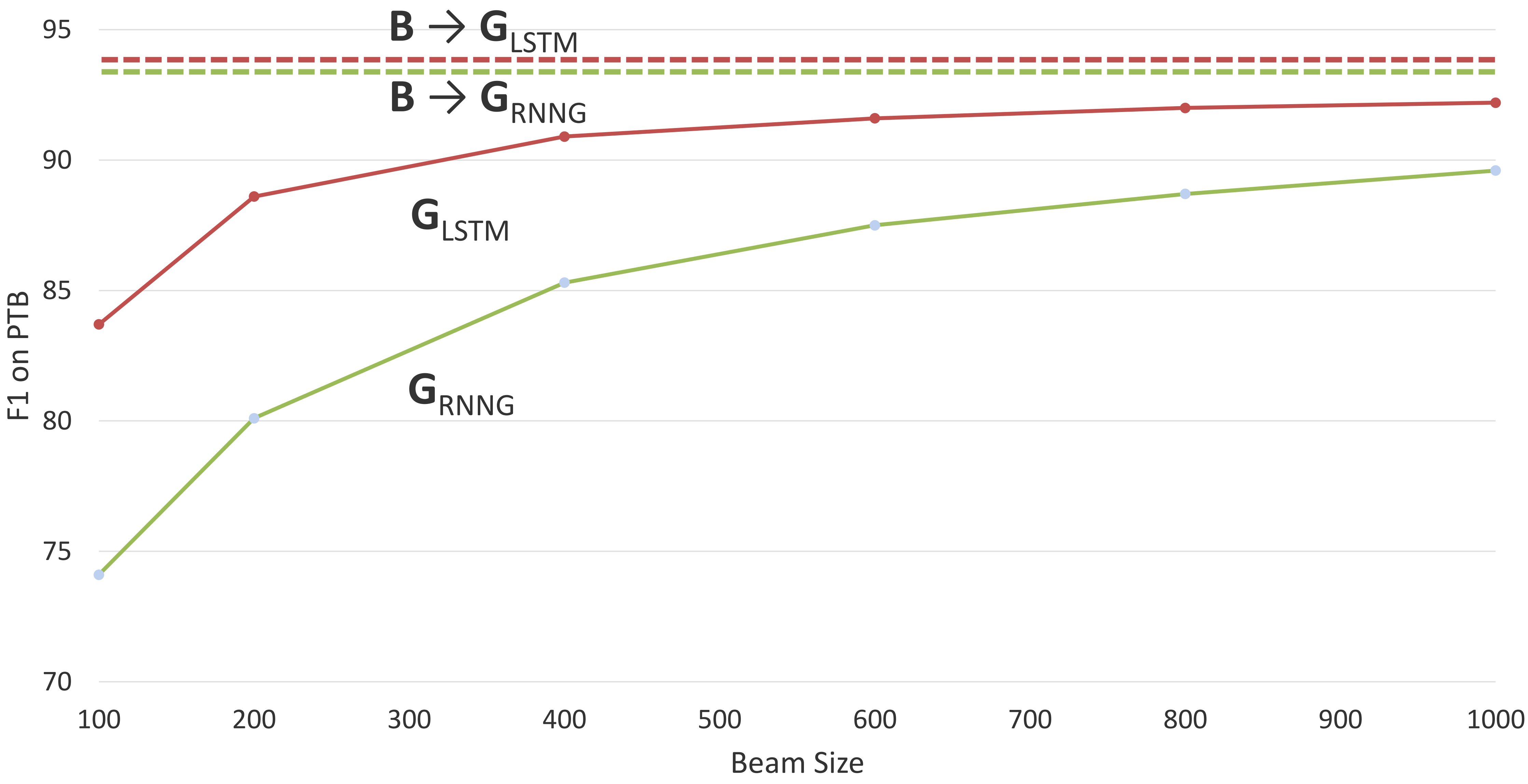


[Roark 2001; Titov and Henderson 2010; Charniak 2010; Buys and Blunsom 2015]

Word-synchronous beam search

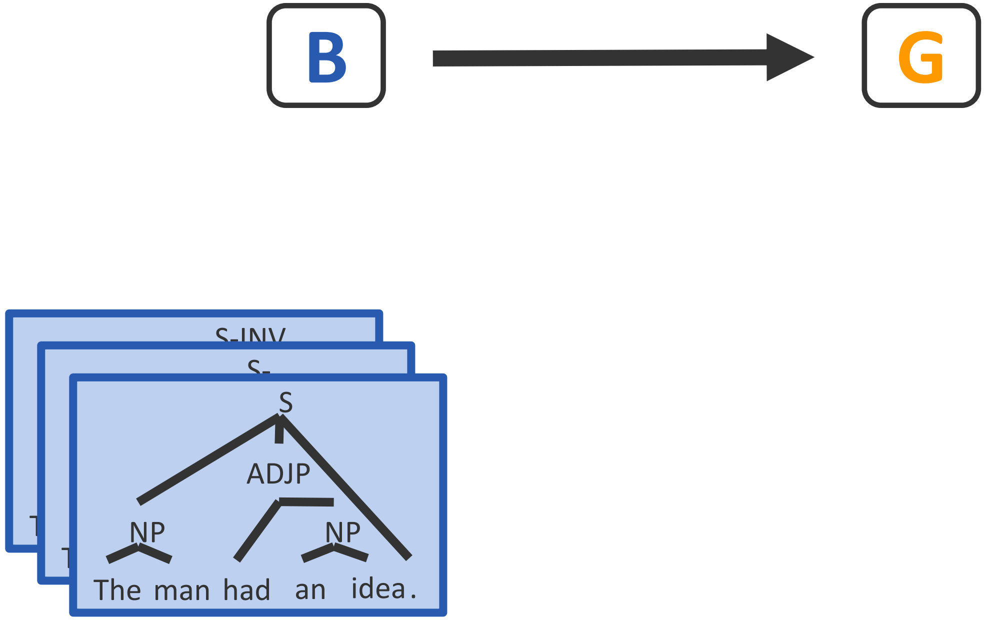


Word-synchronous beam search



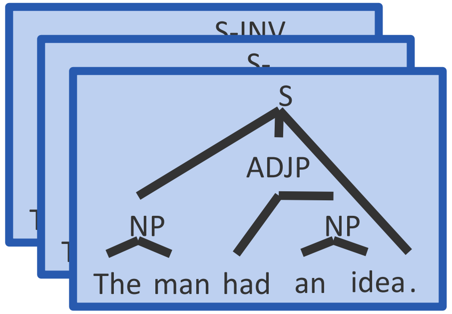
Finding model combination effects

Finding model combination effects



Finding model combination effects

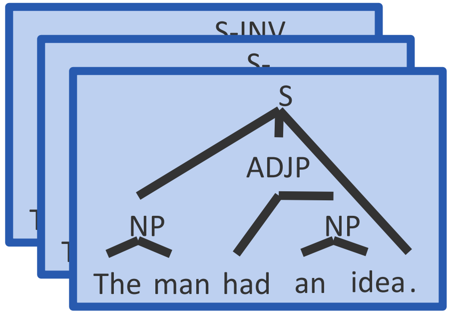
Add G's search proposal to candidate list:



Finding model combination effects

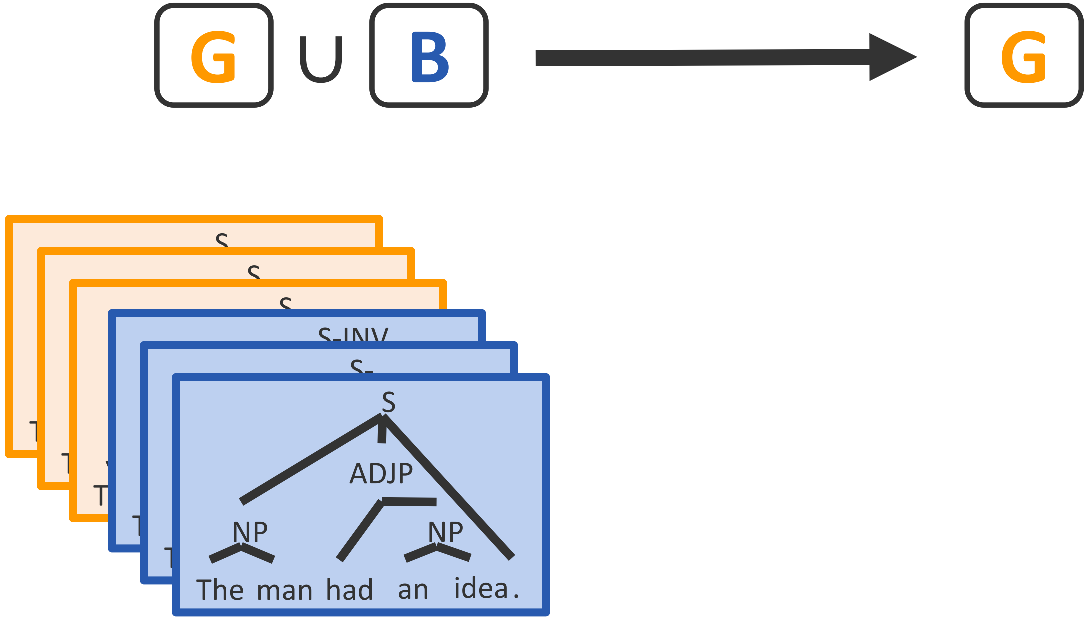
Add G’s search proposal to candidate list:

G ∪ B → G



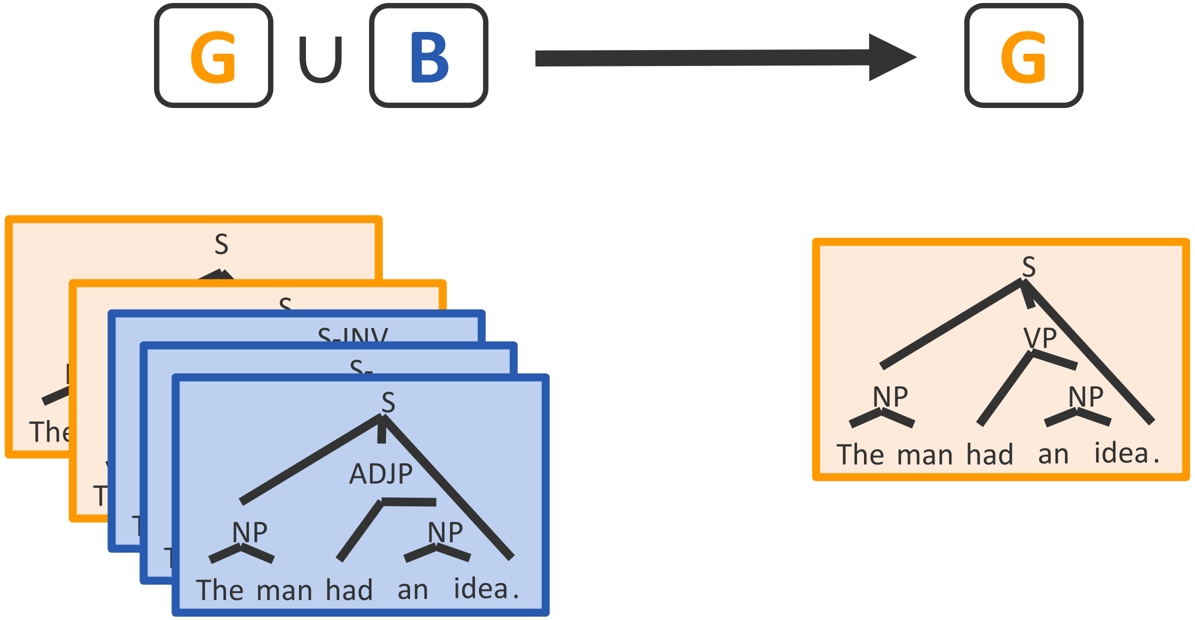
Finding model combination effects

Add G’s search proposal to candidate list:

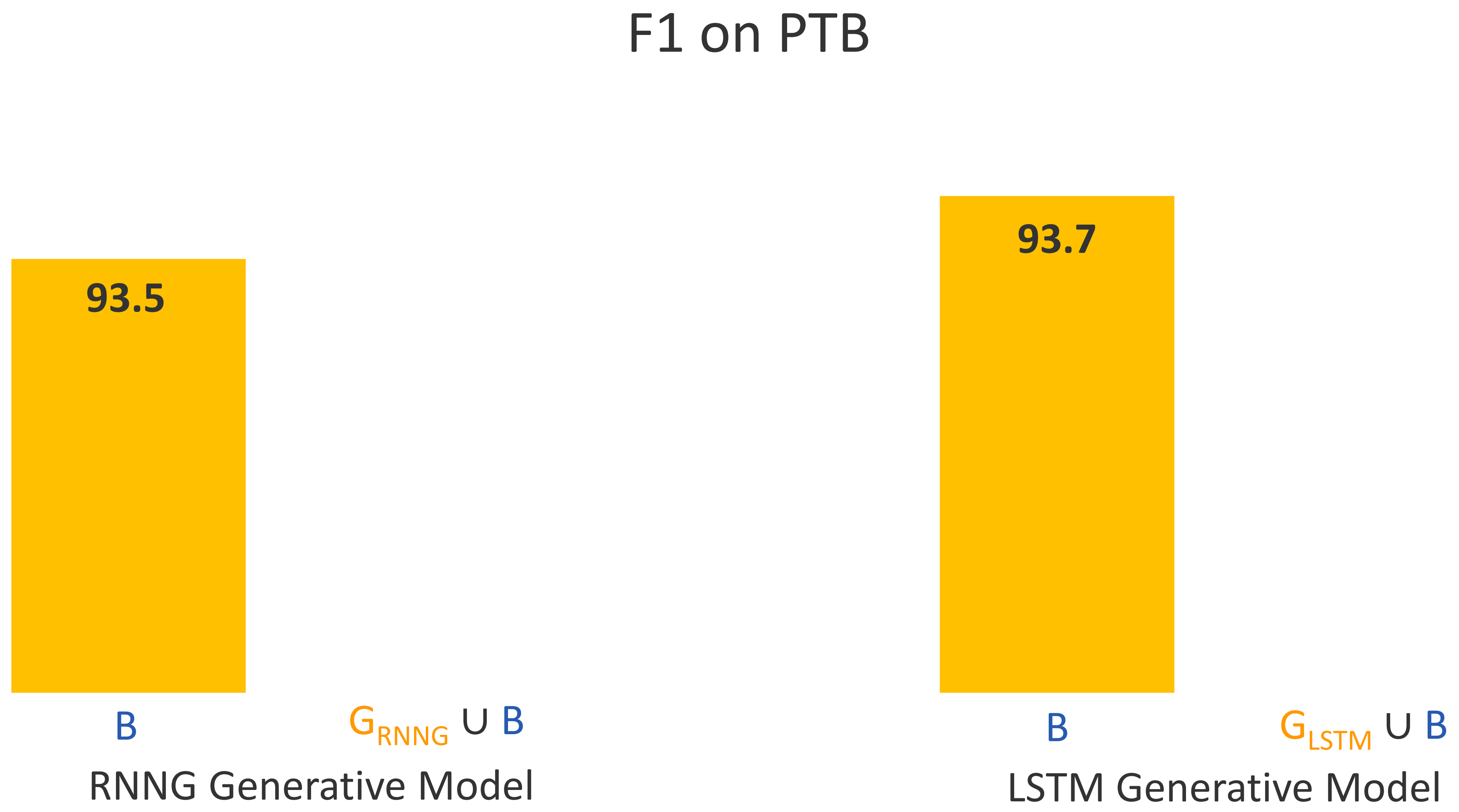


Finding model combination effects

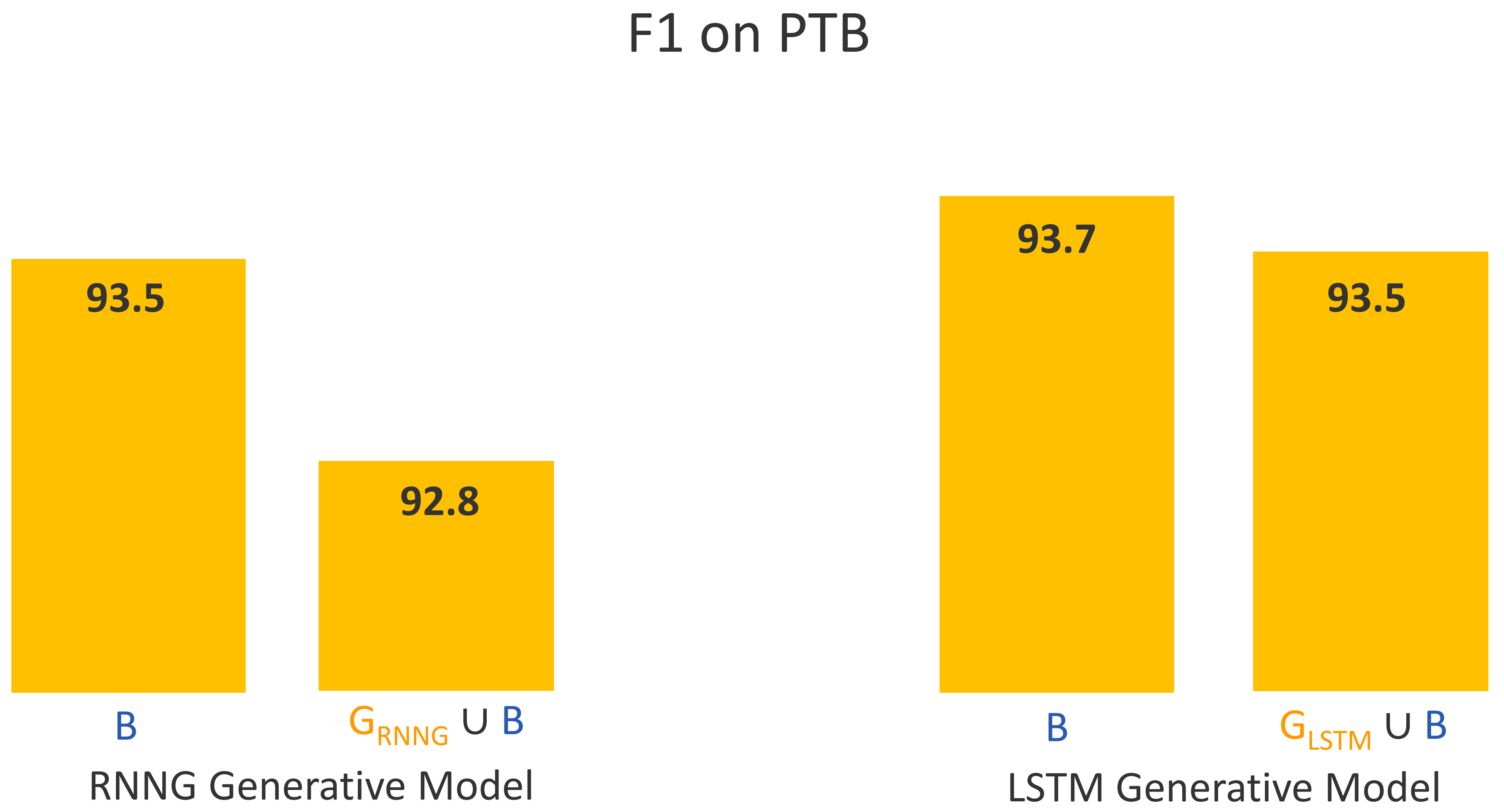
Add G's search proposal to candidate list:



Finding model combination effects



Finding model combination effects



Reranking shows implicit model combination

B hides model errors in **G**

Making model combination explicit

Can we do better by simply combining model scores?

$$\boxed{B} \longrightarrow \boxed{G}$$

$$\boxed{G} \cup \boxed{B} \longrightarrow \boxed{G}$$

$$\log p_G(x, y)$$

Making model combination explicit

Can we do better by simply combining model scores?

$$\boxed{B} \longrightarrow \boxed{G} + \boxed{B}$$

$$\boxed{G} \cup \boxed{B} \longrightarrow \boxed{G} + \boxed{B}$$

$$\log p_G(x, y)$$

Making model combination explicit

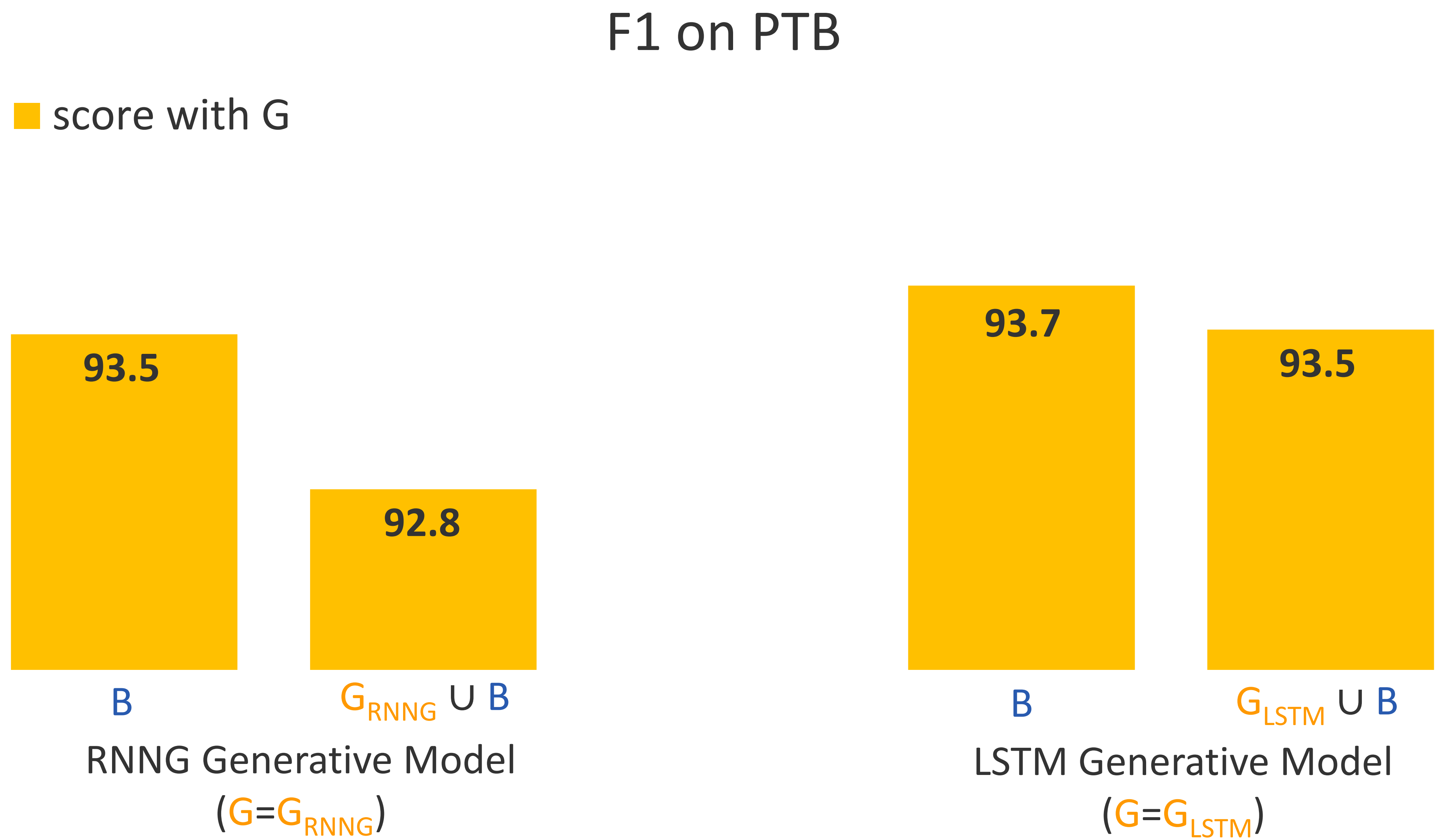
Can we do better by simply combining model scores?

$$\boxed{B} \longrightarrow \boxed{G} + \boxed{B}$$

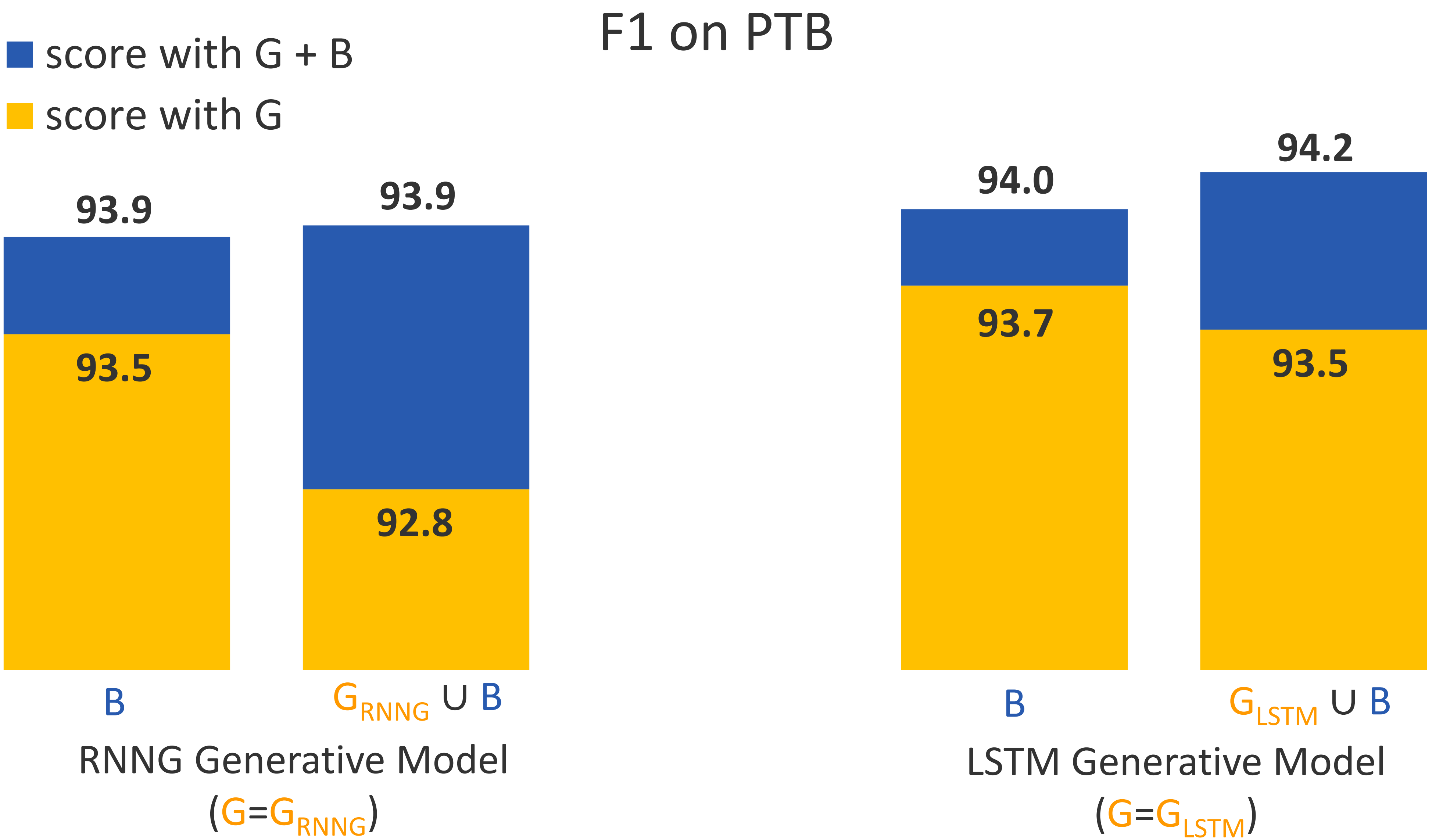
$$\boxed{G} \cup \boxed{B} \longrightarrow \boxed{G} + \boxed{B}$$

$$\lambda \log p_{\boldsymbol{G}}(x, y) + (1 - \lambda) \log p_{\boldsymbol{B}}(y|x)$$

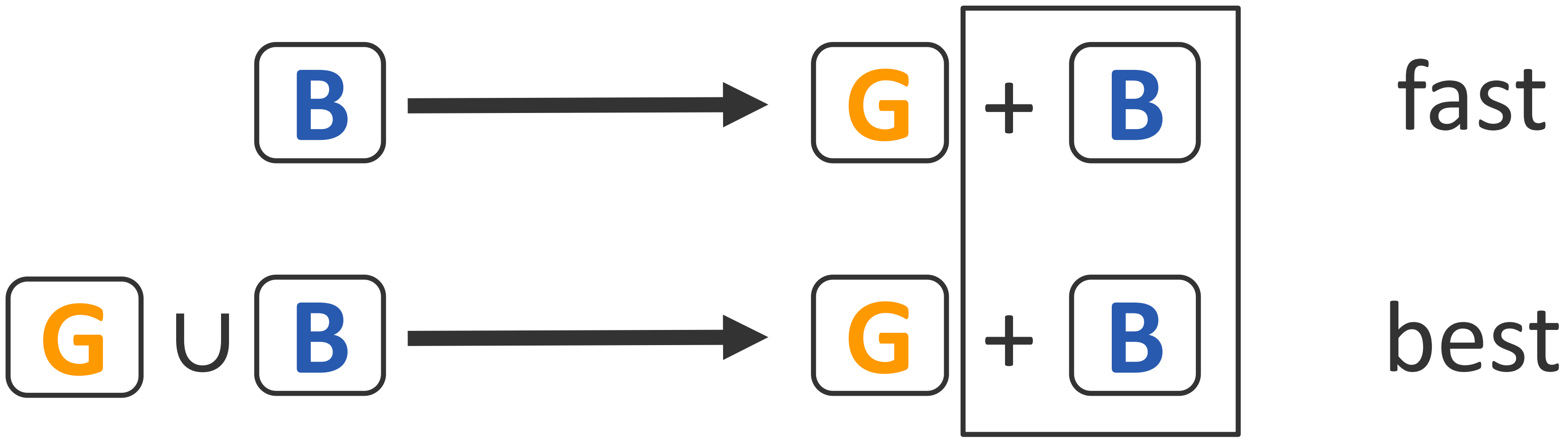
Making model combination explicit



Making model combination explicit



Explicit score combination prevents errors

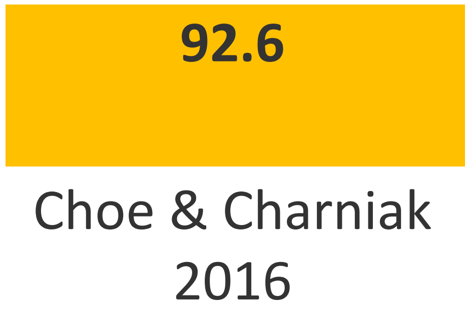


Comparison to past work

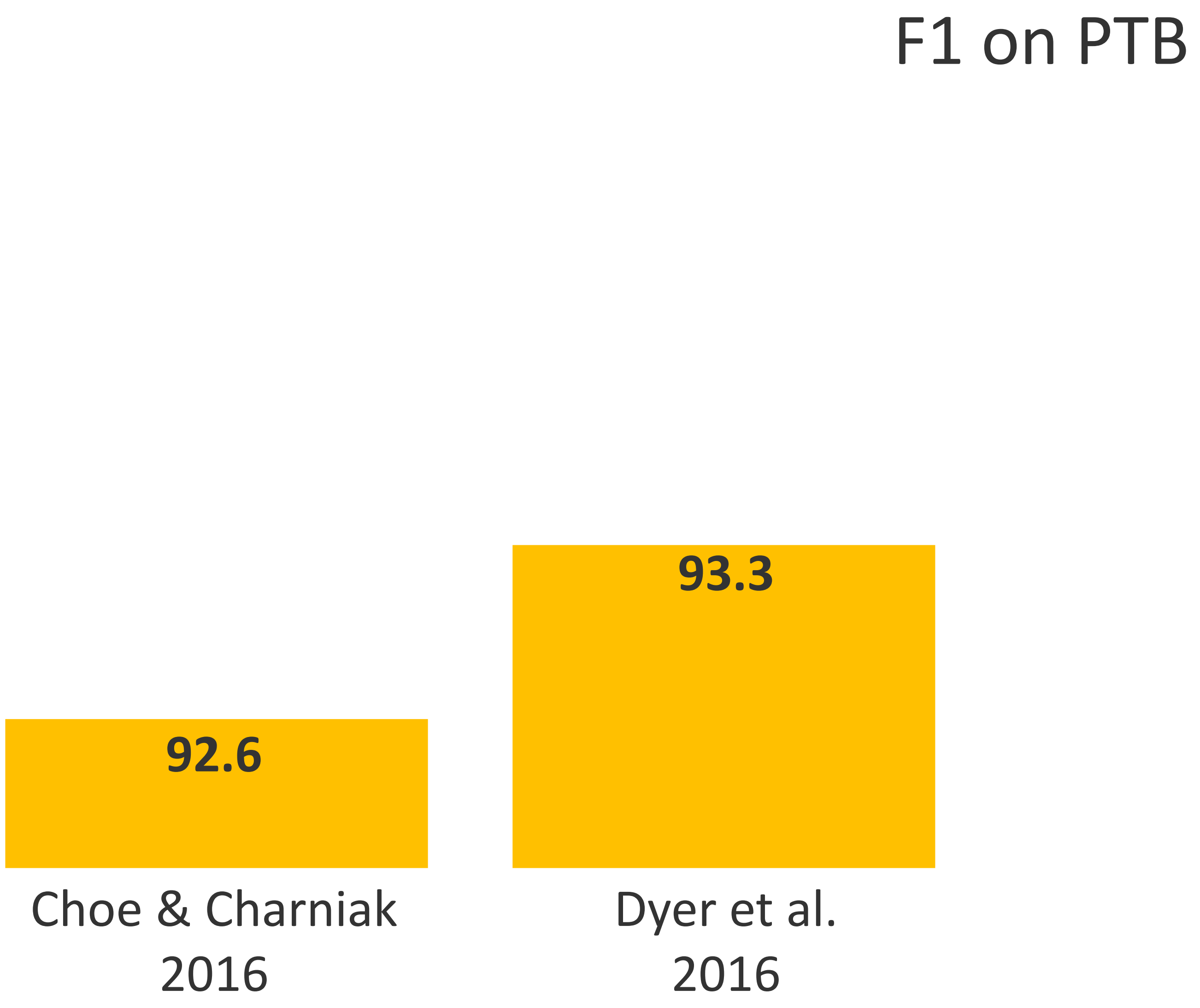
F1 on PTB

Comparison to past work

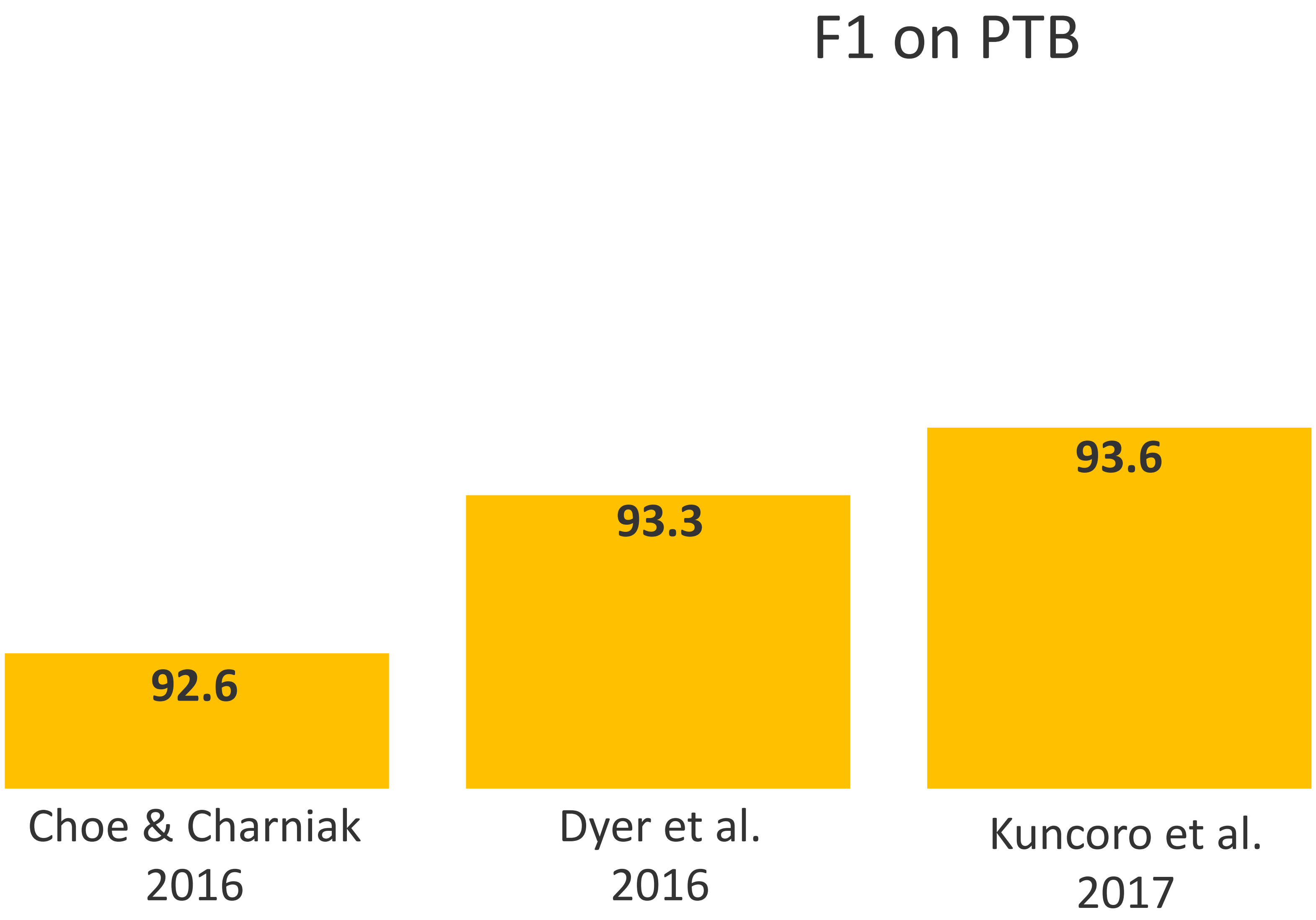
F1 on PTB



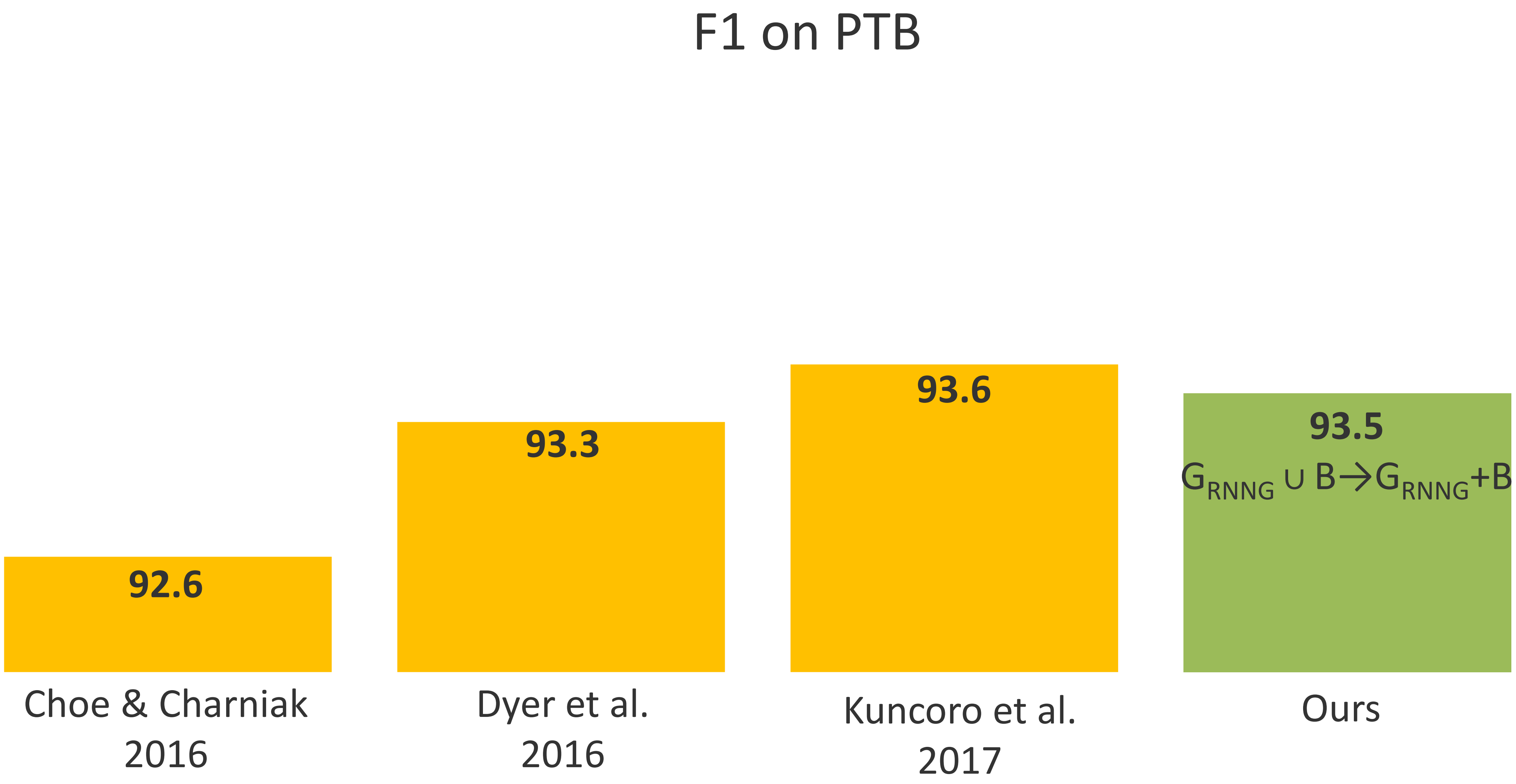
Comparison to past work



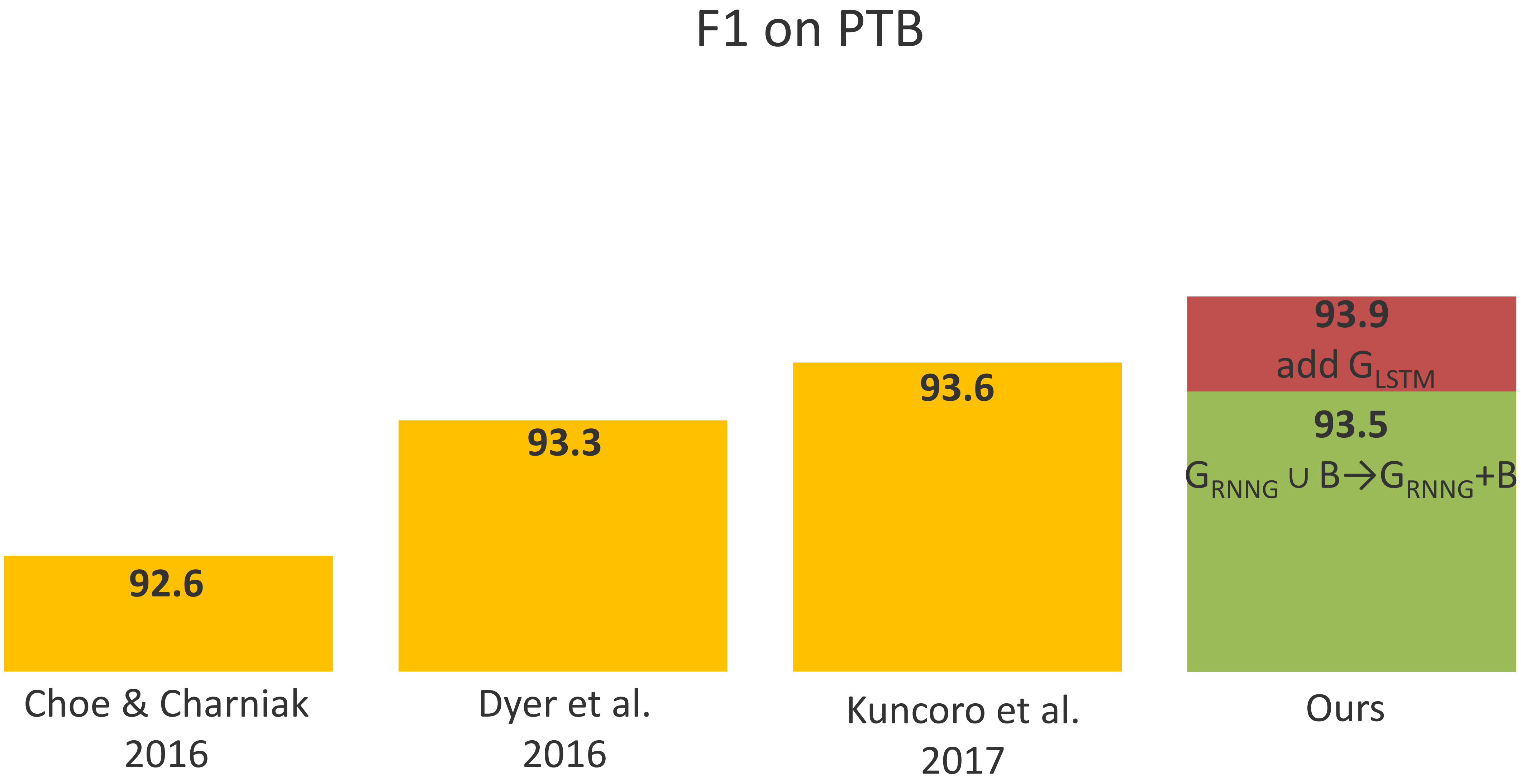
Comparison to past work



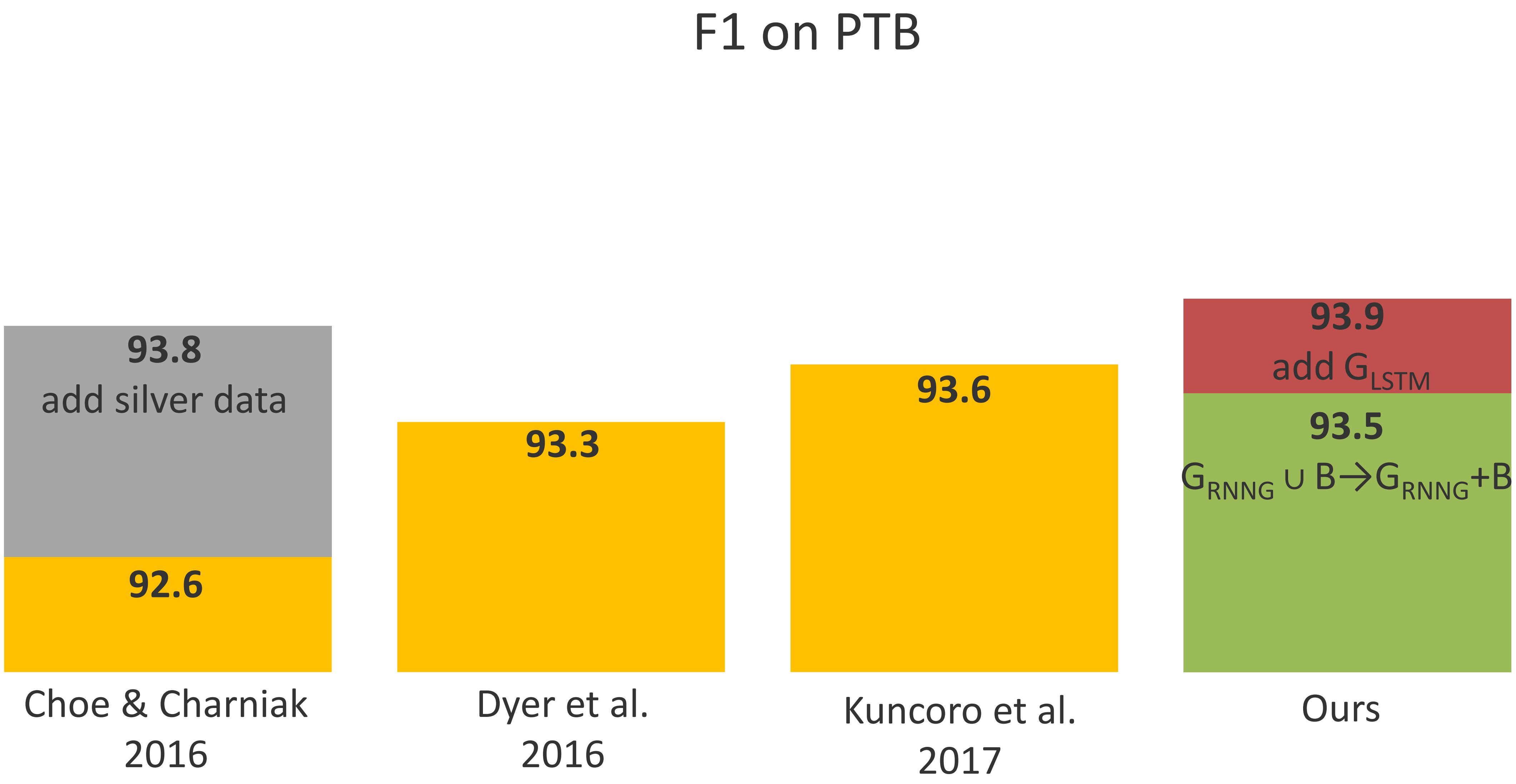
Comparison to past work



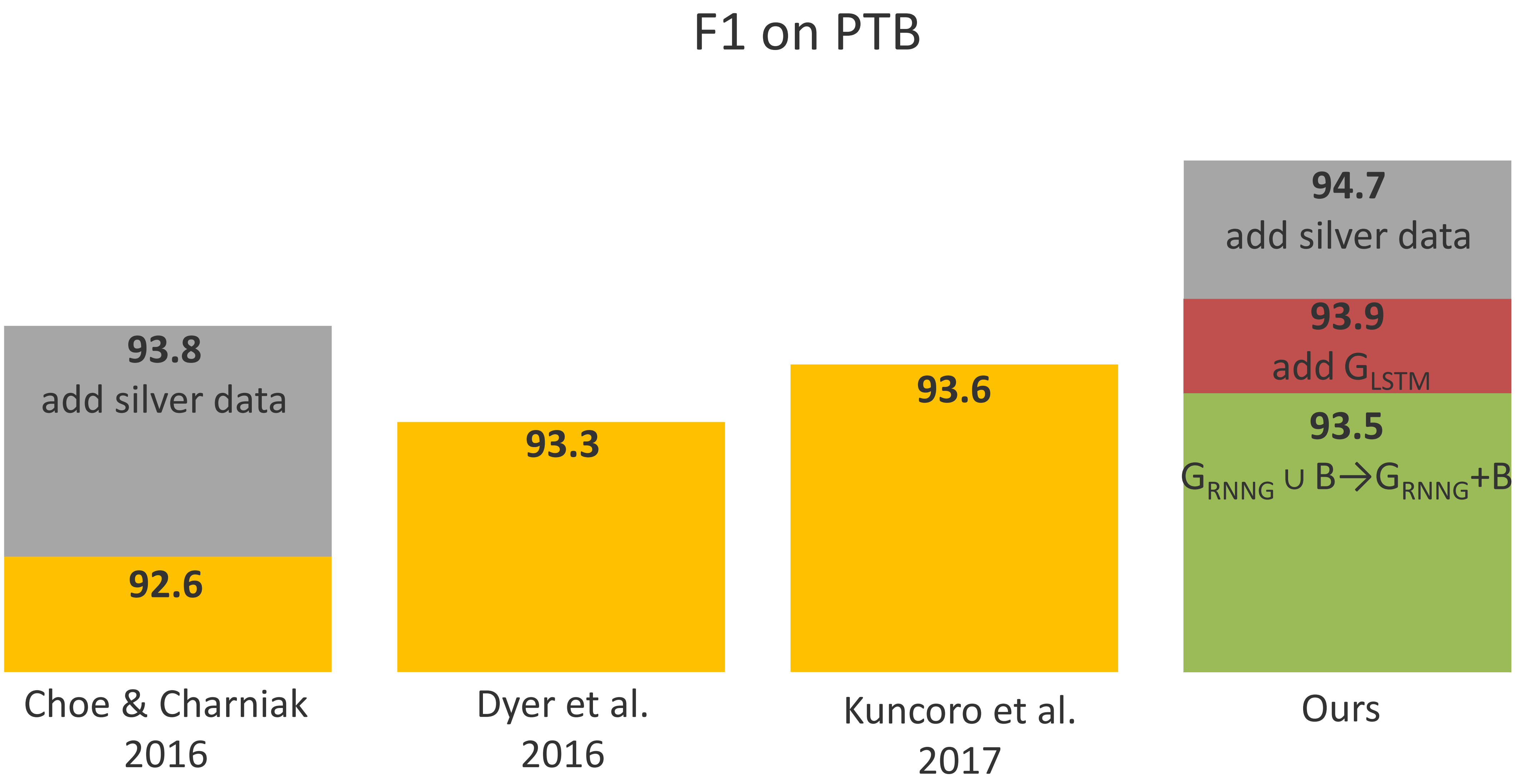
Comparison to past work



Comparison to past work



Comparison to past work



Conclusion

Search procedure for $\boxed{G}$

Conclusion

Search procedure for $\boxed{G}$

(more effective version forthcoming: Stern et al., EMNLP 2017)

Conclusion

Search procedure for G

(more effective version forthcoming: Stern et al., EMNLP 2017)

Found model combination effects in B $\rightarrow$ G

Conclusion

Search procedure for $\boxed{G}$

(more effective version forthcoming: Stern et al., EMNLP 2017)

Found model combination effects in $\boxed{B} \longrightarrow \boxed{G}$

Large improvements from simple, explicit score combination:

$$\boxed{B} \longrightarrow \boxed{G} + \boxed{B}$$

Thanks!