

# Attention-based No-Seg Chinese NER

2018-12-27

# 1. Background

About named entity recognition...

# Background - NER example (character-based 字)

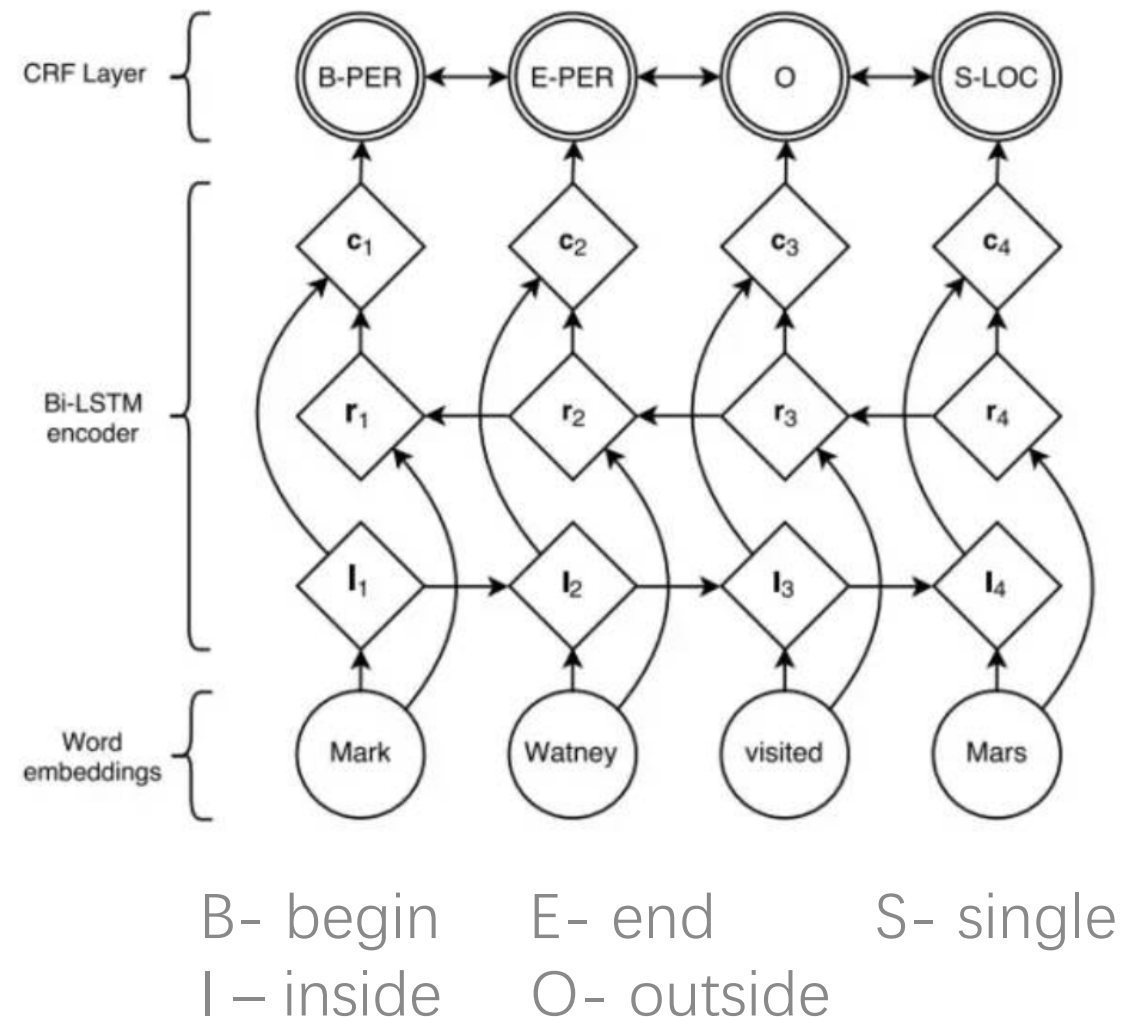
调 查 范 围 涉 及 故 宫 、 古 研 所 、 北 大 清 华  
O O O O O O B-LOC E-LOC O B-ORG I-ORG E-ORG O B-LOC I-LOC I-LOC I-LOC

图 书 馆 、 日 伪 资 料 库 等 二 十 几 家 。  
I-LOC I-LOC E-LOC O S-LOC O O O O O O O O O O

- Using BIOES style, b-begin, i-inside, e-end, o-outside, s-single
- Sequence labeling

# Background – basic deep learning model (word-based)

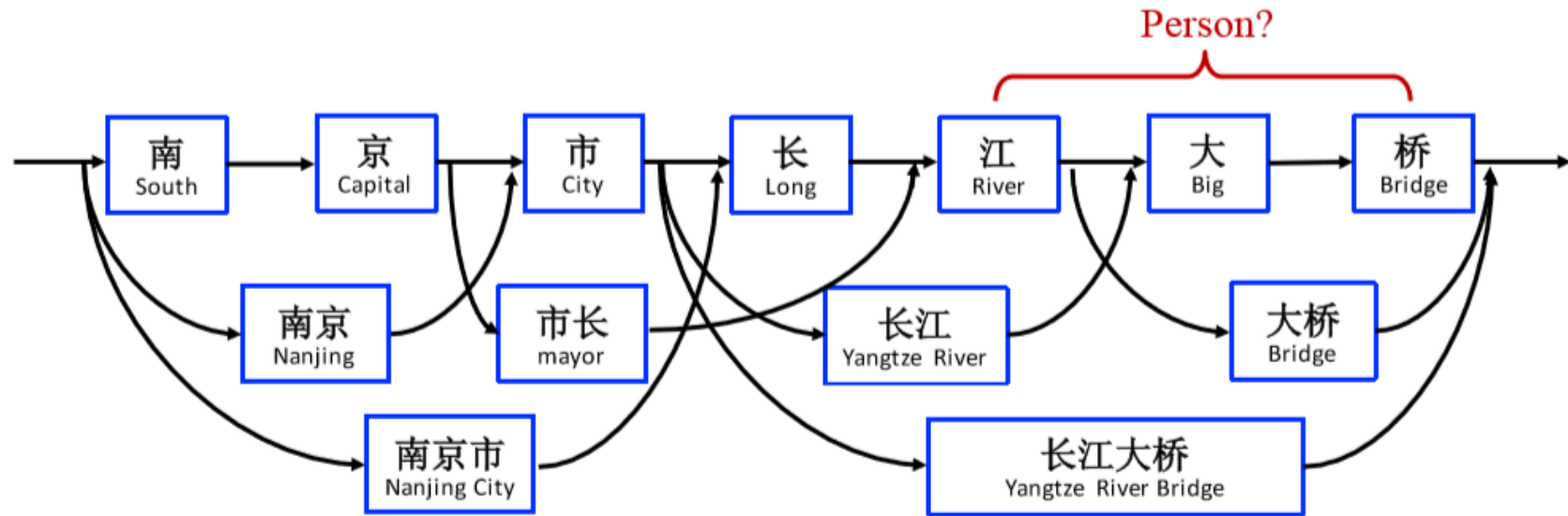
- Bi-LSTM: encoding information at each time stamp.
- CRF(conditional random field): learning rules between tags in a sequence.  
(B-xx should be followed by I-xx or E-xx)



## 2. Challenges

With Chinese NER...

# Challenges



南京/市长/江大桥 or 南京市/长江大桥

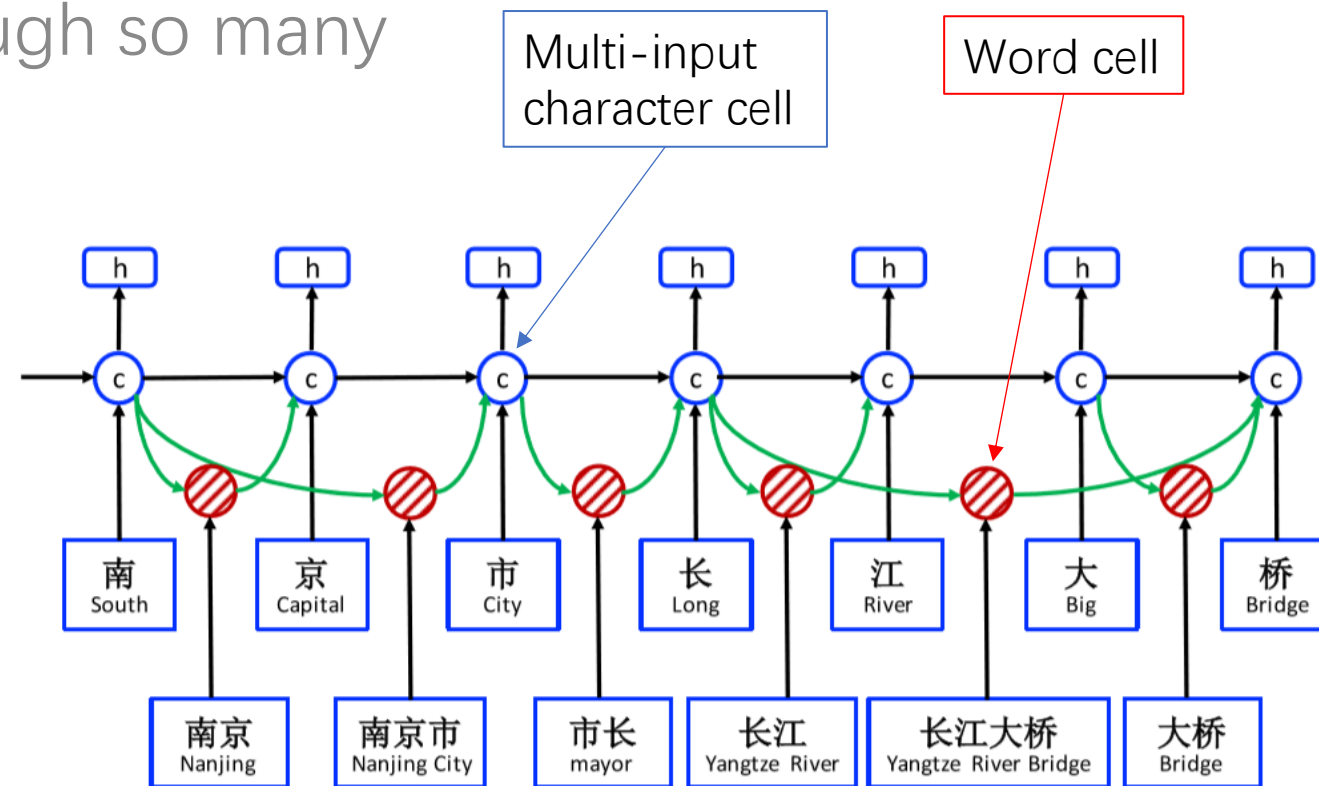
- **No natural word boundaries.**
- If use vanilla character-based bi-LSTM (feed character embeddings):
  - lack word semantic information
- If use word-based bi-LSTM (feed word embeddings)
  - word segmentation error propagation, resolve entities

# Goal

*Better integrate word-level  
information and character-  
level information.*

# Lattice LSTM (SOTA)

- Redundant and ineffective  
LSTM cells are costly,  
Forgetting and updating through so many  
word cells and character cells.
- Loss of info:  
Information of a word only  
arrives at the ending character  
of this word, without influence  
on characters before it.





# 3. Proposed Model

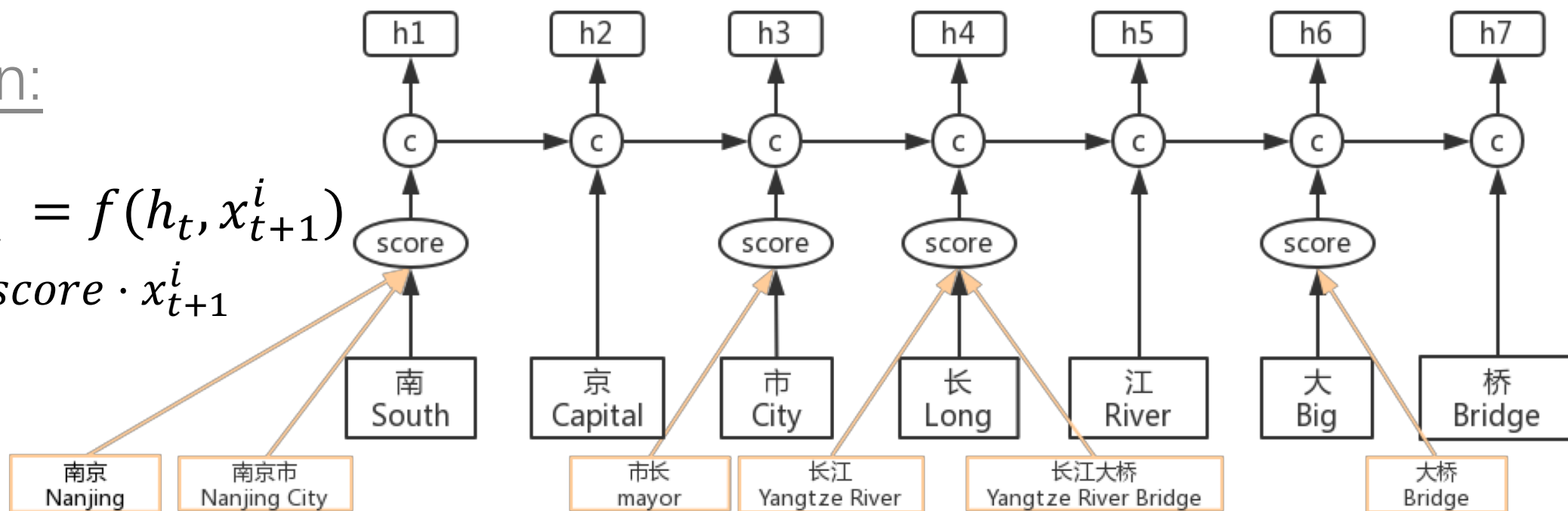
Inspired by *Chinese NER Using Lattice LSTM, 2018 ACL*

# Proposed Model 1

attention:

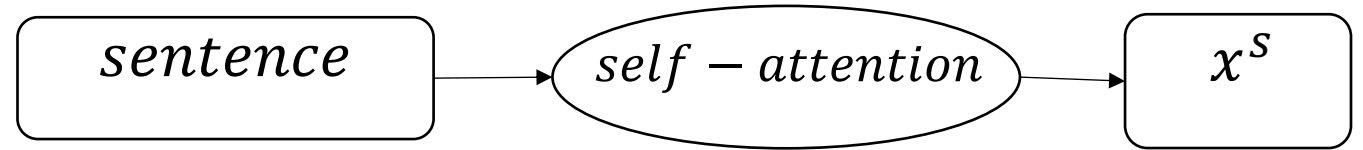
$$score_{t+1}^i = f(h_t, x_{t+1}^i)$$

$$x'_{t+1} = \sum score \cdot x_{t+1}^i$$



\*the words come from auto segmentation on a huge corpus

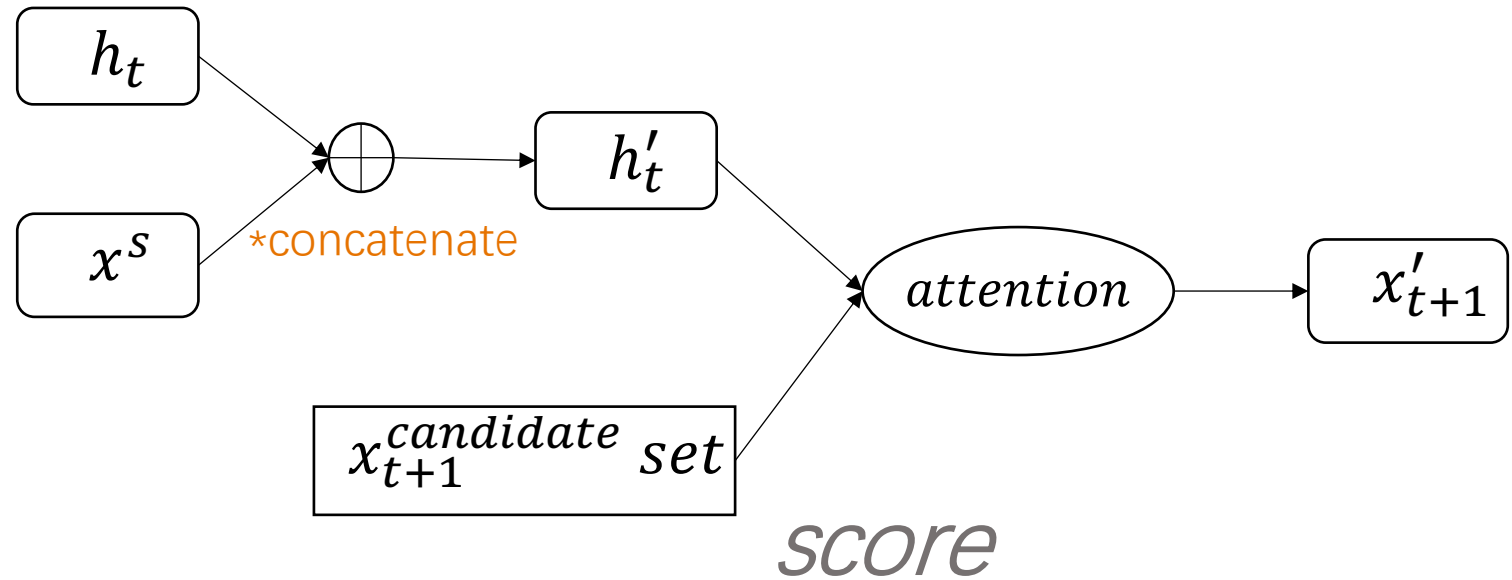
# Proposed Model 2



attention:

$$score_{t+1}^i = f(h'_t, x_{t+1}^i)$$

$$x'_{t+1} = \sum score \cdot x_{t+1}^i$$

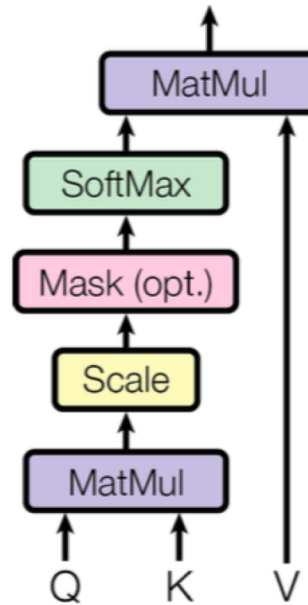


# Proposed Model 2

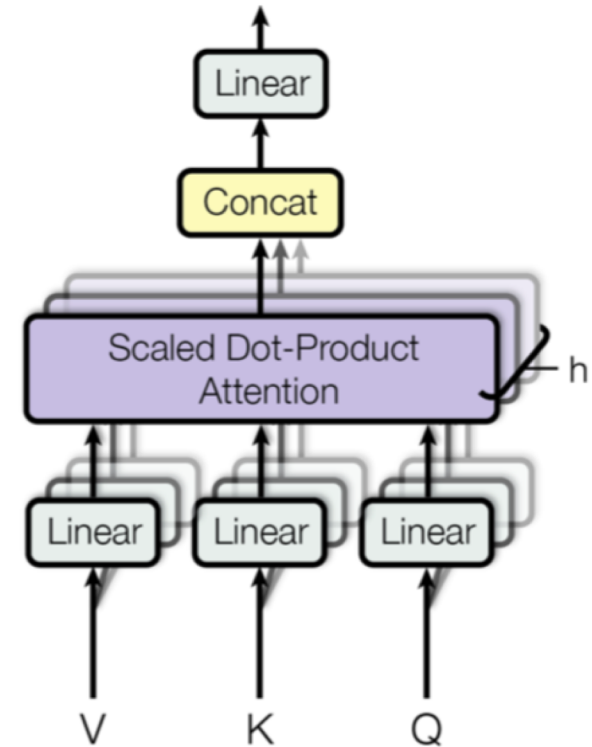
Self-attention<sub>(multi-head)</sub>:

$$Q = K = V$$

Scaled Dot-Product Attention



Multi-Head Attention



$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

# Possible Attempts

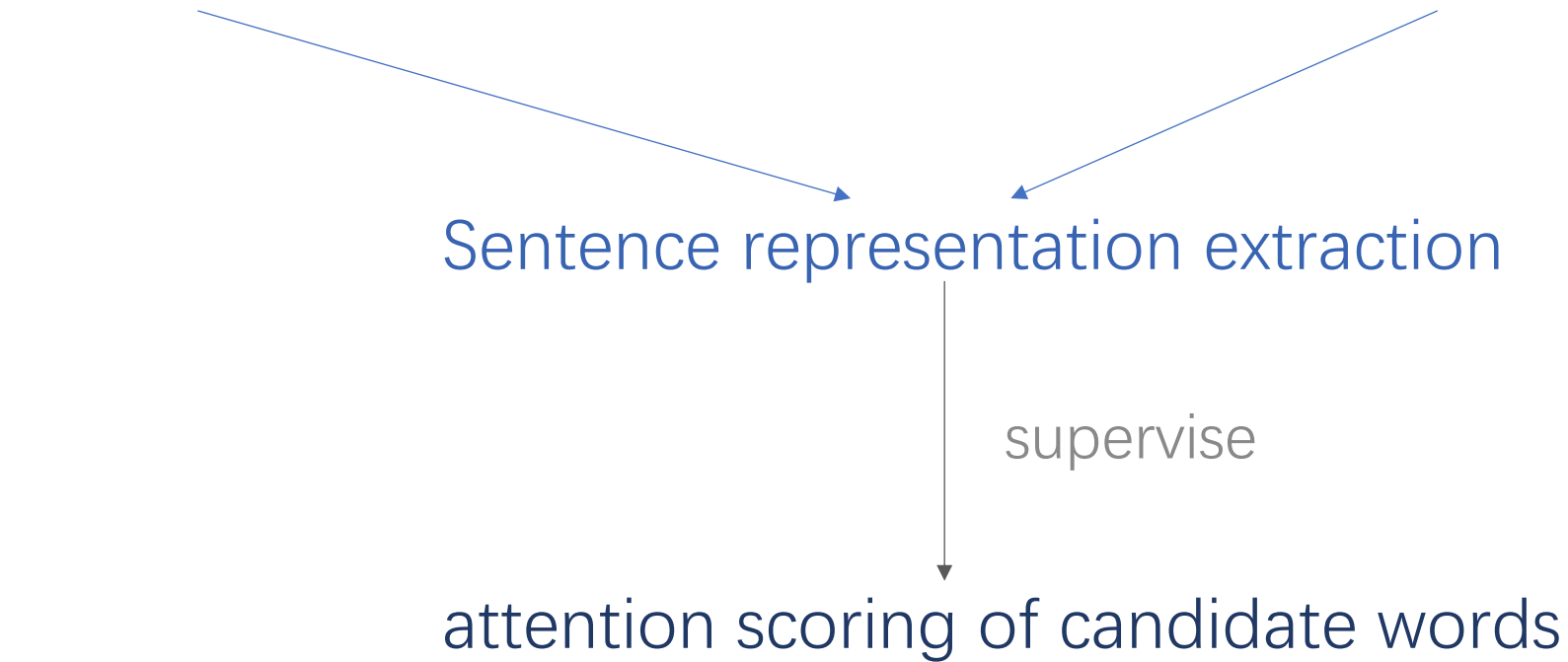
self-attention

vanilla character-based bi-LSTM

Sentence representation extraction

supervise

attention scoring of candidate words



# Possible Attempts

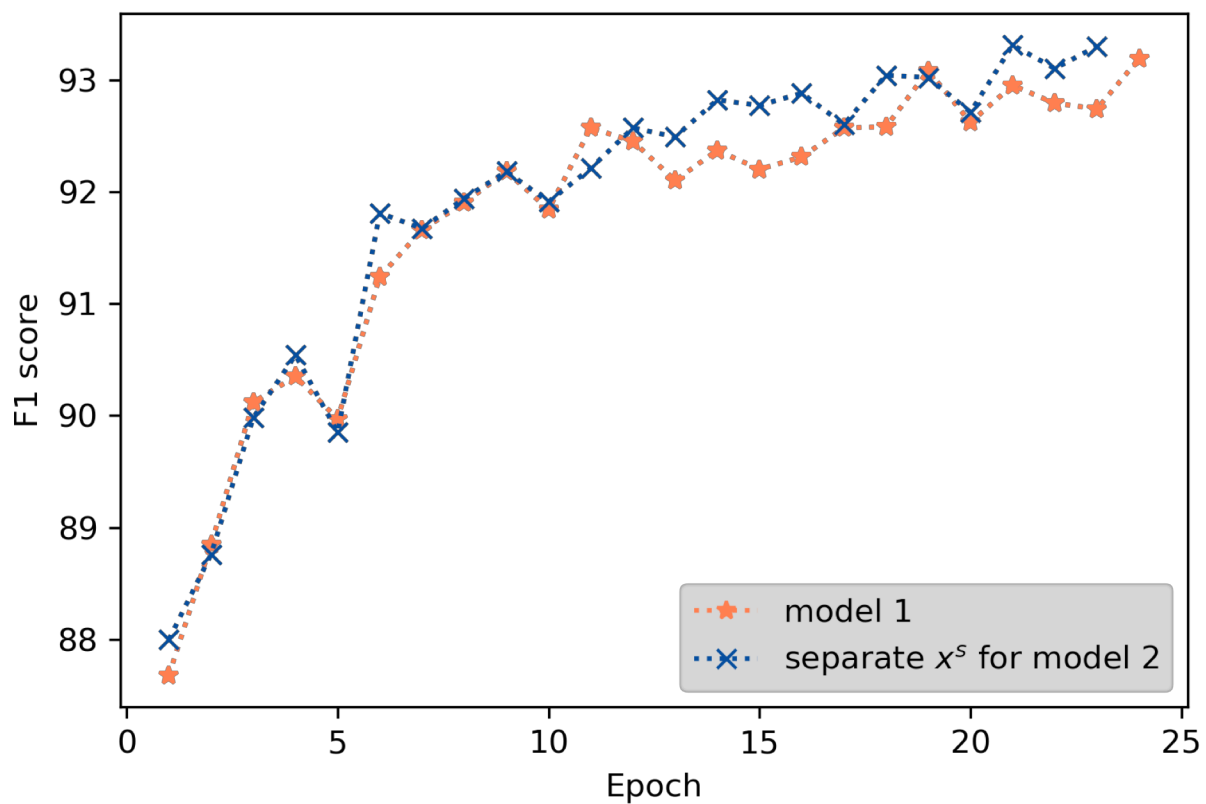
1. Transformation from word embedding space to character embedding space (failed)

$$score^c \cdot x_{t+1}^c + \sum_i score^{w_i} W_{w \rightarrow c} \cdot x_{t+1}^{w_i}$$

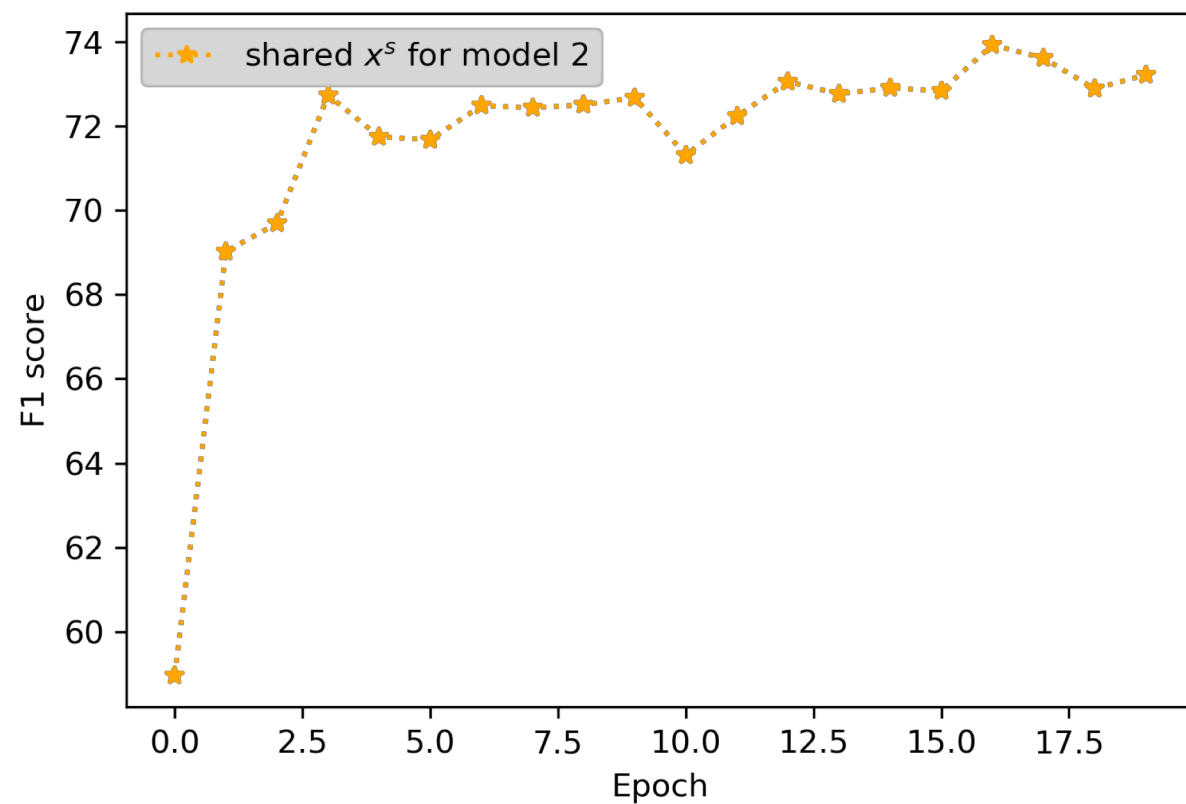
2. Sentence embedding: shared or separate
3. Hidden states from char lstm as sentence embeddings, attention over hidden states.
4. Different ways to combine  $x^s$  and  $h_t$  : concat, sum, mul, ...

# Results so far...

Performance on MSRA dataset



Performance on OntoNote dataset



# Results so far...

This part is deprecated.

The current best F1 for MSRA is:

Model 1 - 93.59

Model 2 – 93.50

Datasets	Models	P	R	F1
MSRA	Lattice	93.57	<b>92.79</b>	93.18
	Model 1	93.95	92.44	93.20
	Model 2	<b>94.51</b>	92.15	<b>93.33</b>
OntoNotes	Lattice	<b>76.35</b>	71.56	73.88
	Model 2	74.56	<b>73.30</b>	<b>73.93</b>



Thanks.