

# Character-based Neural Networks for Sentence Pair Modeling



THE OHIO STATE UNIVERSITY

Wuwei Lan and Wei Xu

## Introduction

- Sentence pair modeling is critical for [paraphrase identification](#), [question answering](#), [natural language inference](#) and etc.
- Various neural models achieved state-of-the-art performance by using [pretrained word embeddings](#), however they have [poor coverage](#) in domain (e.g., social media) with [high OOV ratio](#).
- We explored [character-based neural networks](#) for sentence pair modeling, which is [more challenging](#) than individual sentence modeling: similarly spelled words with completely different meanings could introduce error (e.g., *ware* and *war*).

## Example for Sentence Pair Modeling

Paraphrase task: given a sentence pair, predict whether they imply the same meaning. Sample from **Twitter URL** corpus [Lan et.al 2017]:

Why Samsung abandoned its popular phone , the Galaxy Note 7	Paraphrase
What Finally Made #Samsung Let Go of Its Galaxy #SamsungNote7 Phone	
Celebrate Hamilton Day , an even better math holiday than Pi Day	Non-paraphrase
So , did anyone actually celebrate #HamiltonDay on Oct 16?	

## Pairwise Word Interaction Model (PWIM) [He et.al 2016]

### 1. Context modeling:

$$\begin{aligned}\vec{h}_i &= LSTM^f(w_i, \vec{h}_{i-1}) \\ \vec{h}_i &= LSTM^b(w_i, \vec{h}_{i+1}) \\ \vec{h}_i &= [\vec{h}_i^f, \vec{h}_i^b] \\ \vec{h}_i^+ &= \vec{h}_i^f + \vec{h}_i^b\end{aligned}$$

### 2. Pairwise word interaction:

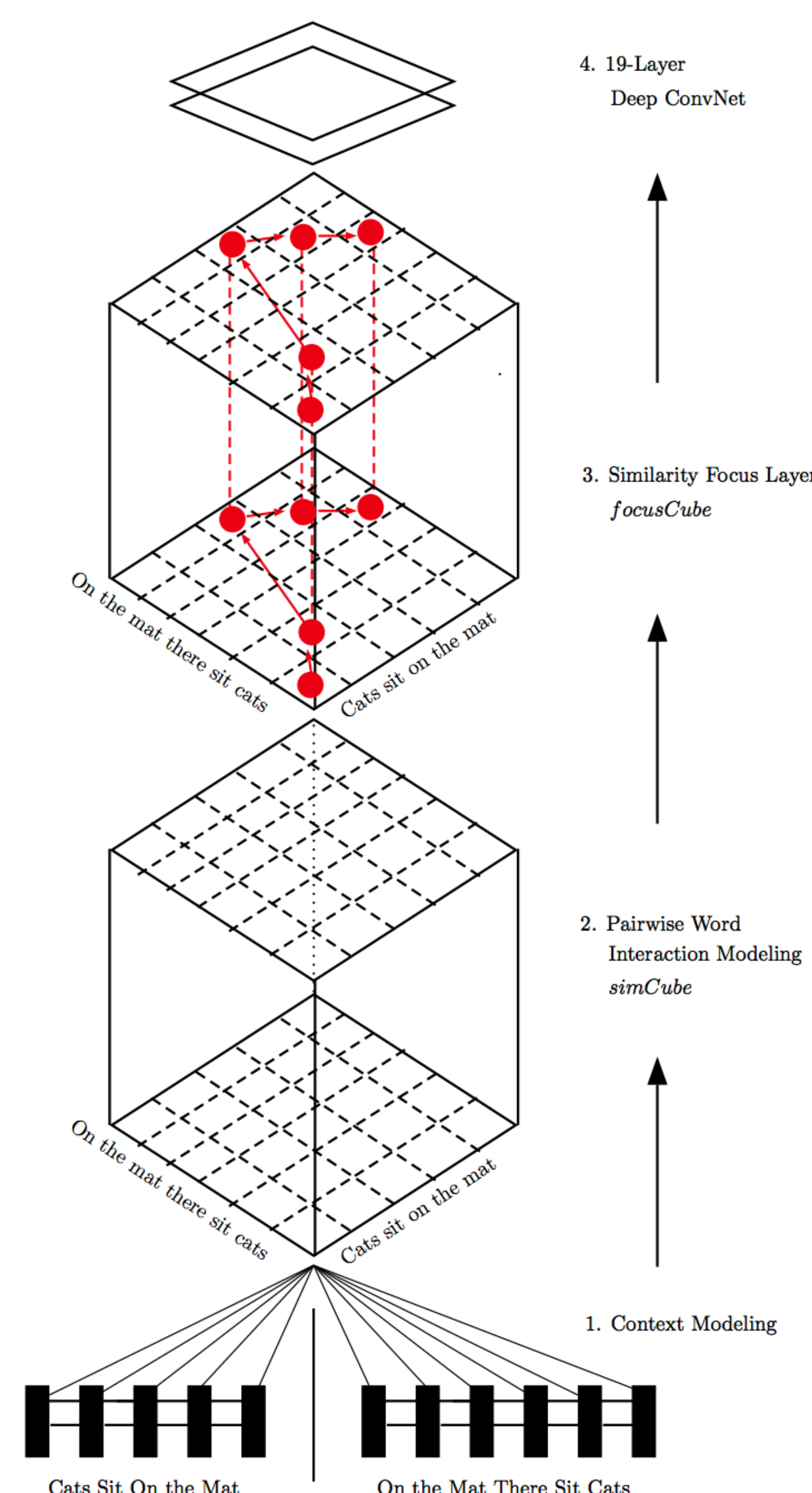
$$D(\vec{h}_i, \vec{h}_j) = [\cos(\vec{h}_i, \vec{h}_j), \\ L2Euclid(\vec{h}_i, \vec{h}_j), \\ DotProduct(\vec{h}_i, \vec{h}_j)].$$

### 3. Similarity focus:

sorting the interaction values and selecting top ranked pairs

### 4. Aggregation and prediction:

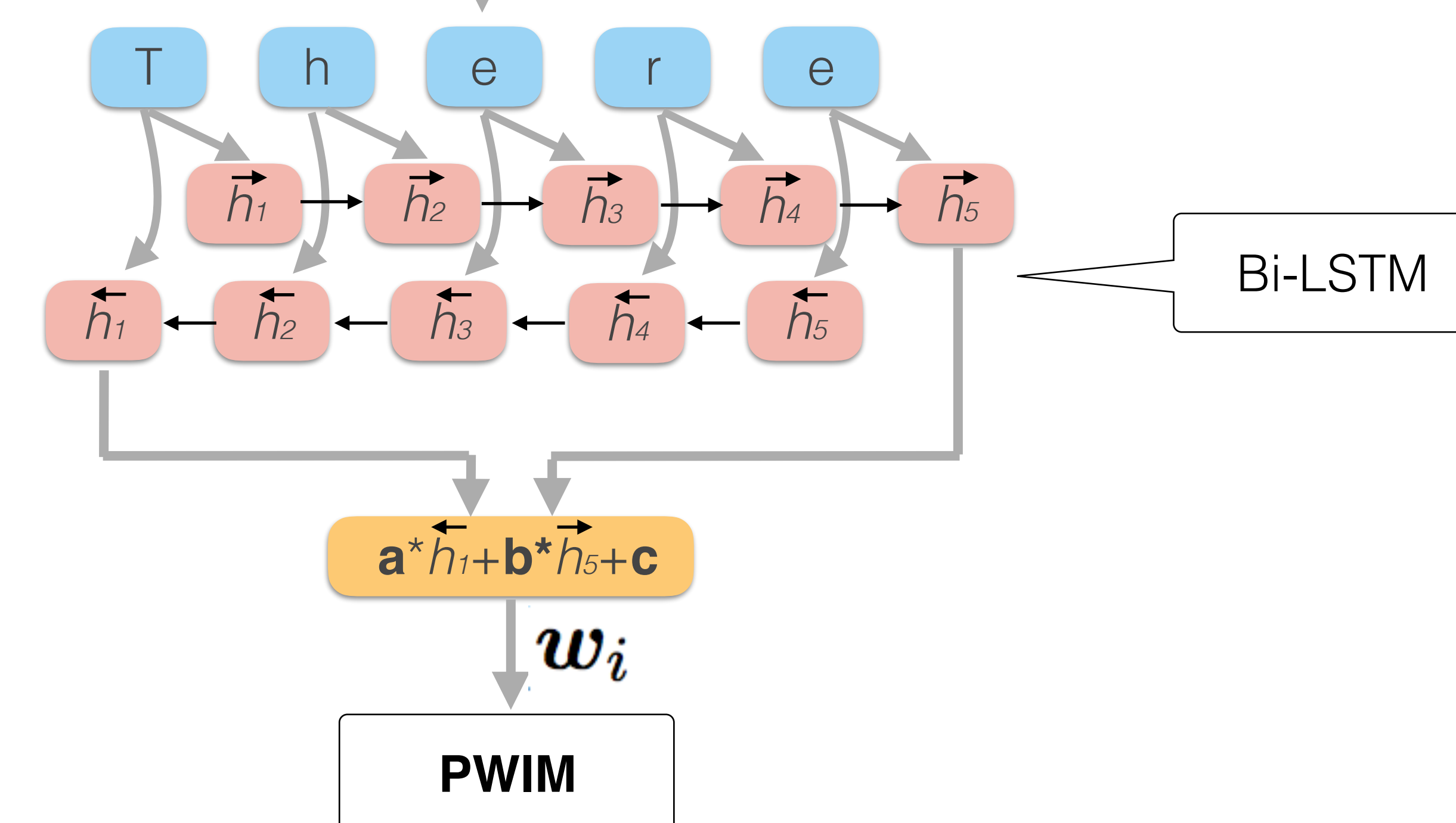
19-layer deep ConvNet



## Embedding Subwords in PWIM

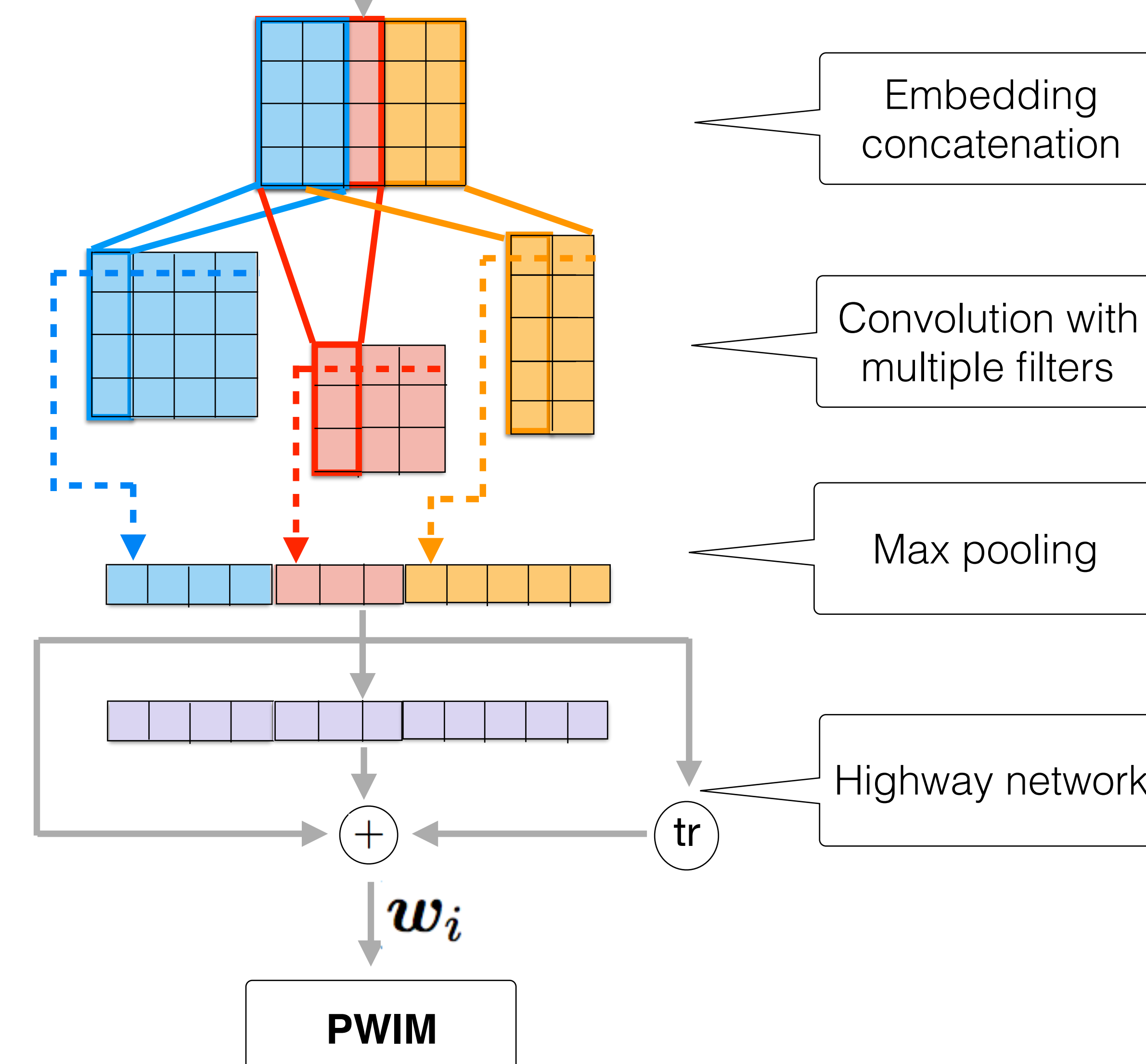
### 1. Char RNN [Ling et.al 2015]

On the Mat **There** Sit Cats



### 2. Char CNN [Kim et.al 2016]

On the Mat **There** Sit Cats



## Conclusion

- ✓ Pretrained word embedding is not a necessity for sentence pair modeling.
- ✓ Subword models without any pretraining achieved new SOTA results in Twitter URL and PIT-2015.
- ✓ Multitask LM can improve subword model performance by injecting semantic information.

## Source Code

<https://github.com/lanwuwei/Subword-PWIM>

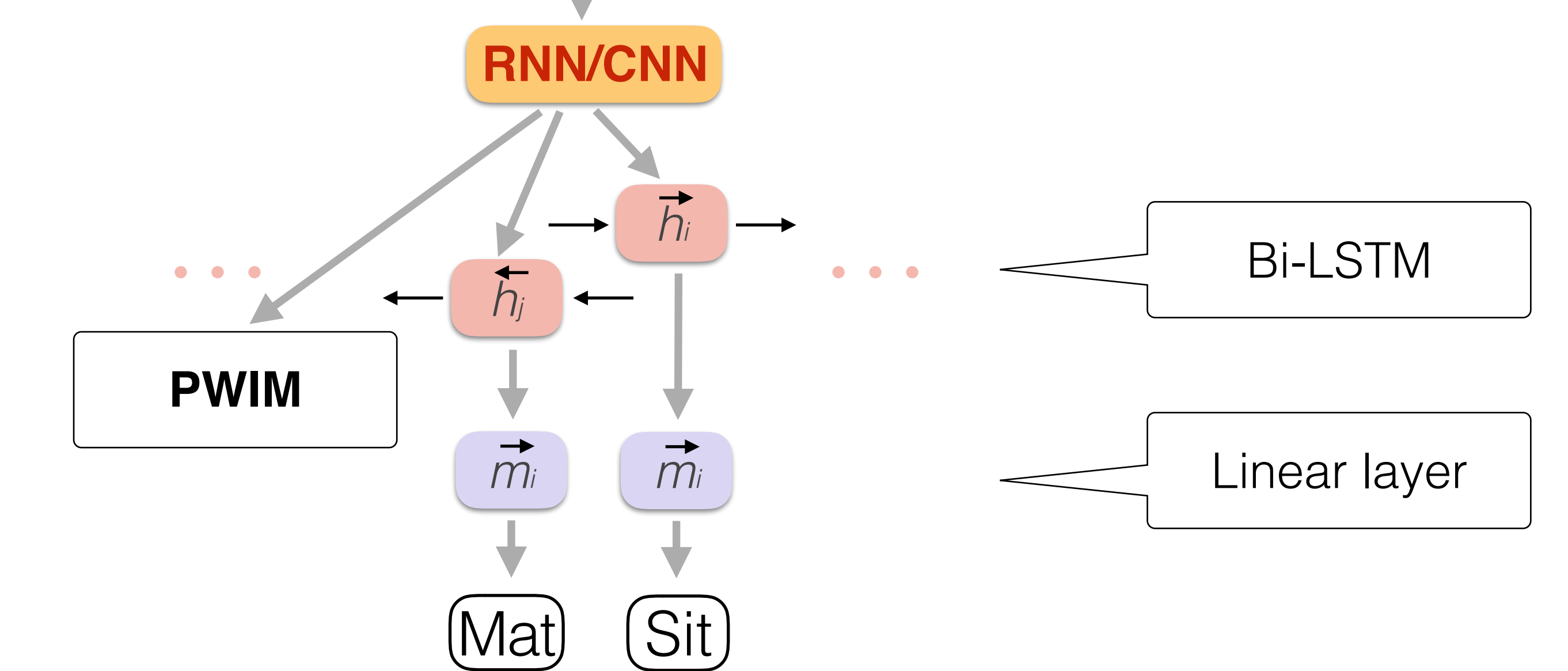


## See Also

- Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. A continuously growing dataset of sentential paraphrases. In EMNLP 2017
- Wuwei Lan and Wei Xu. Neural Network Models for Paraphrase Identification, Semantic Textual Similarity, Natural Language Inference, and Question Answering. In COLING 2018

## Language Modeling Objective (Multitask)

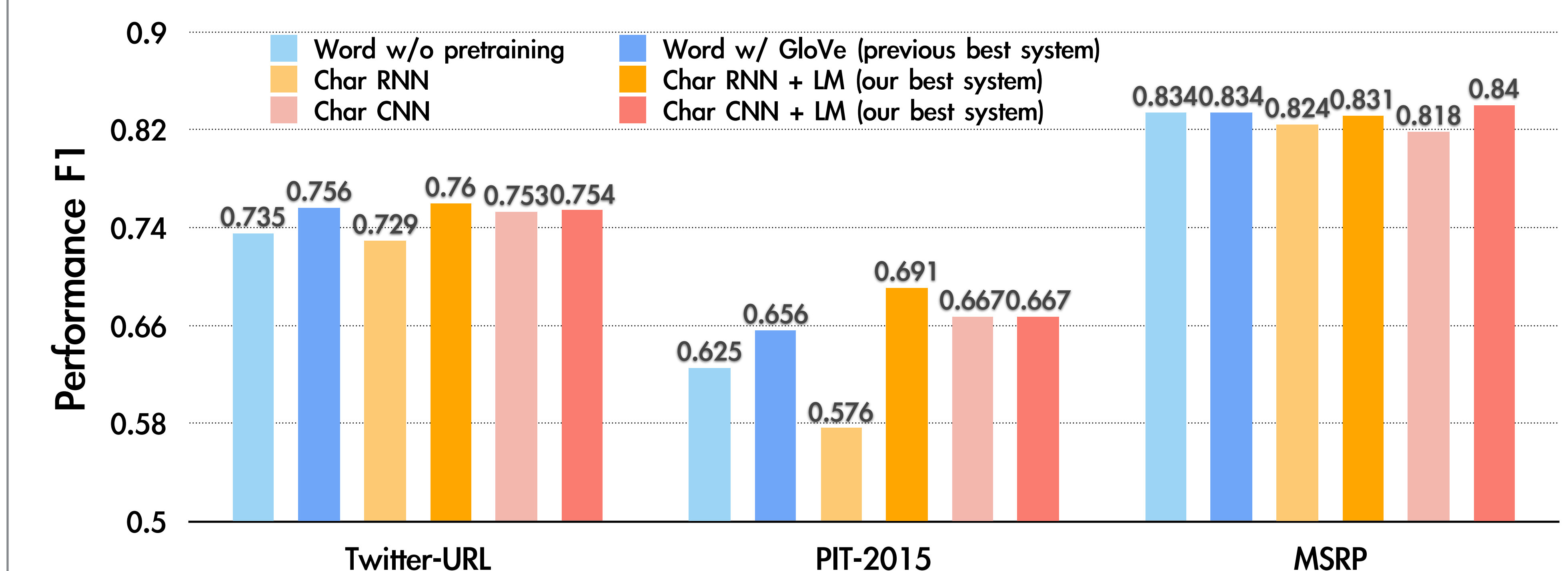
On the Mat **There** Sit Cats



The forward LSTM predicts the **next word** and the backward LSTM predicts the **previous word**. The log-likelihood loss for both language models is added to the training objective:

$$\begin{aligned}\vec{E} &= E + \gamma(\vec{E} + \overleftarrow{E}) \\ \vec{E} &= - \sum_{t=1}^{T-1} \log(P(w_{t+1} | \vec{m}_t)) \quad \overleftarrow{E} = - \sum_{t=2}^T \log(P(w_{t-1} | \overleftarrow{m}_t))\end{aligned}$$

## Experiments



We performed experiments on three benchmark datasets for paraphrase identification: **Twitter URL** (social media/news, OOV ratio 31.5%), **PIT-2015** (social media, OOV ratio 13.7%) and **MSRP** (news, OOV ratio 9.0%).

Model	INV Words		OOV Words	
	any	walking	#airport	brexit
Word	anything	walk	salomon	bollocks
	anyone	running	363	missynistic
	other	dead	#trumpdchotel	patriarchy
Subword	analogy	waffling	@atairport	grexit
	nay	slagging	#dojreport	bret
	away	scaling	#macbookpro	juliet
Subword + LM	any1	warming	airport	#brexit
	many	wagging	#airports	brit
	ang	waging	rapport	ofbrexit

Nearest neighbors of word (subword, subword + LM) vectors under cosine similarity in Twitter-URL dataset.