Modeling Local Dependence in Natural Language with Multi-Channel Recurrent Neural Networks

Chang Xu, Weiran Huang, Hongwei Wang, Gang Wang and Tie-Yan Liu

Nankai University, Tsinghua University, Shanghai Jiao Tong University, Microsoft Research Asia

Presented by: Chang Xu

Content

- Motivation
    - Background of Modeling Natural Sentences
    - Challenges in Capturing Semantic Structure Information
- Our Method: Multi-Channel Recurrent Neural Networks
    - Capturing Rich Patterns with Multiple Channels
    - Aggregating Patterns by an Attention Module
- Experimental Results
- Analysis
    - Case Study and Visualization
    - Performance on Long Sentences
    - Impact of Model Size
- Conclusion

Modeling Natural Sentences

- Structure Information is Essential
  - Natural languages exhibit strong local structures in terms of semantics such as phrases.

    E.g. We must find the missing document at all costs.
  - Phrase structures are important for understanding the meaning of sentences

- Conventional Recurrent Neural Networks
  - Usually treat each token in a sentence ***uniformly and equally***
  - May ***miss the rich semantic structure*** information of a sentence.

Challenges in Capturing Semantic Structure Information

- Requiring Flexibility
  - There are diverse word dependence patterns
  - *Flexible and learnable structure modeling method* is preferred than *predefined connections or fixed topology*.

- Hard to Parameterize
  - The local structures and word dependence patterns in sentences are discrete symbols rather than regular learnable model parameters.
  - It is non-trivial to capture and parameterize them.

Content

- Motivation
  - Background of Modeling Natural Sentences
  - Challenges in Capturing Semantic Structure Information
- Our Method: Multi-Channel Recurrent Neural Networks
  - Capturing Rich Patterns with Multiple Channels
  - Aggregating Patterns by an Attention Module
- Experimental Results
- Analysis
  - Case Study and Visualization
  - Performance on Long Sentences
  - Impact of Model Size
- Conclusion

Multi-Channel Recurrent Neural Networks (MC-RNN)

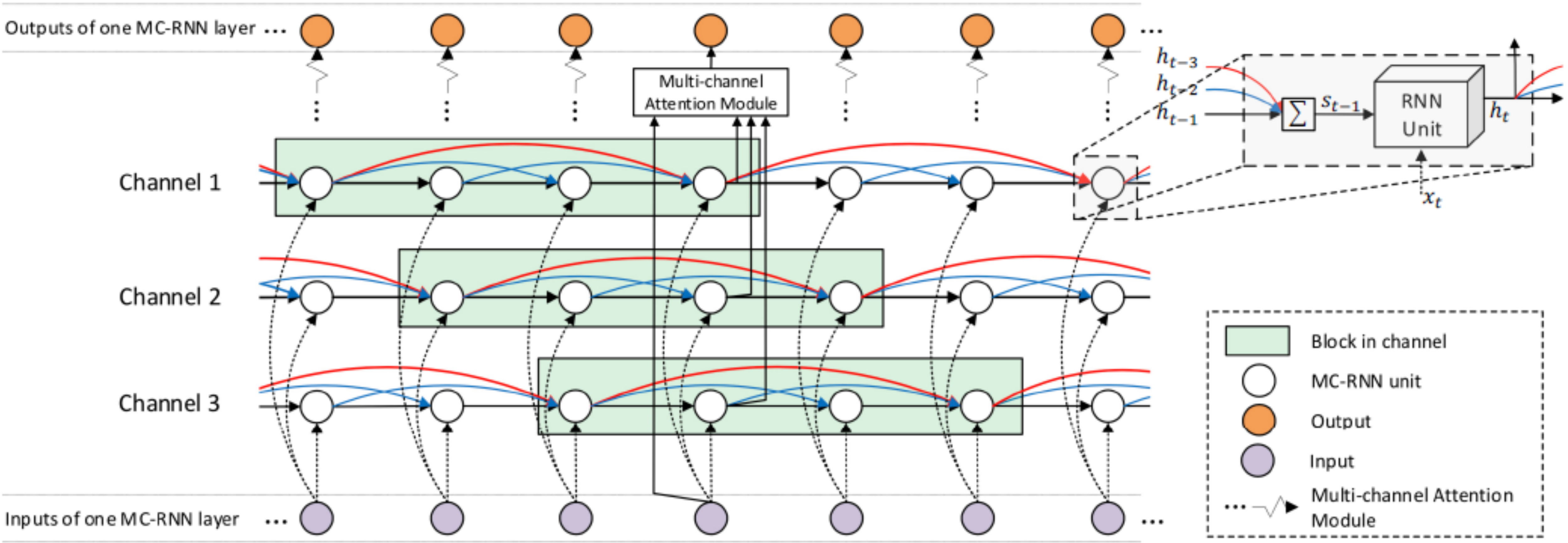
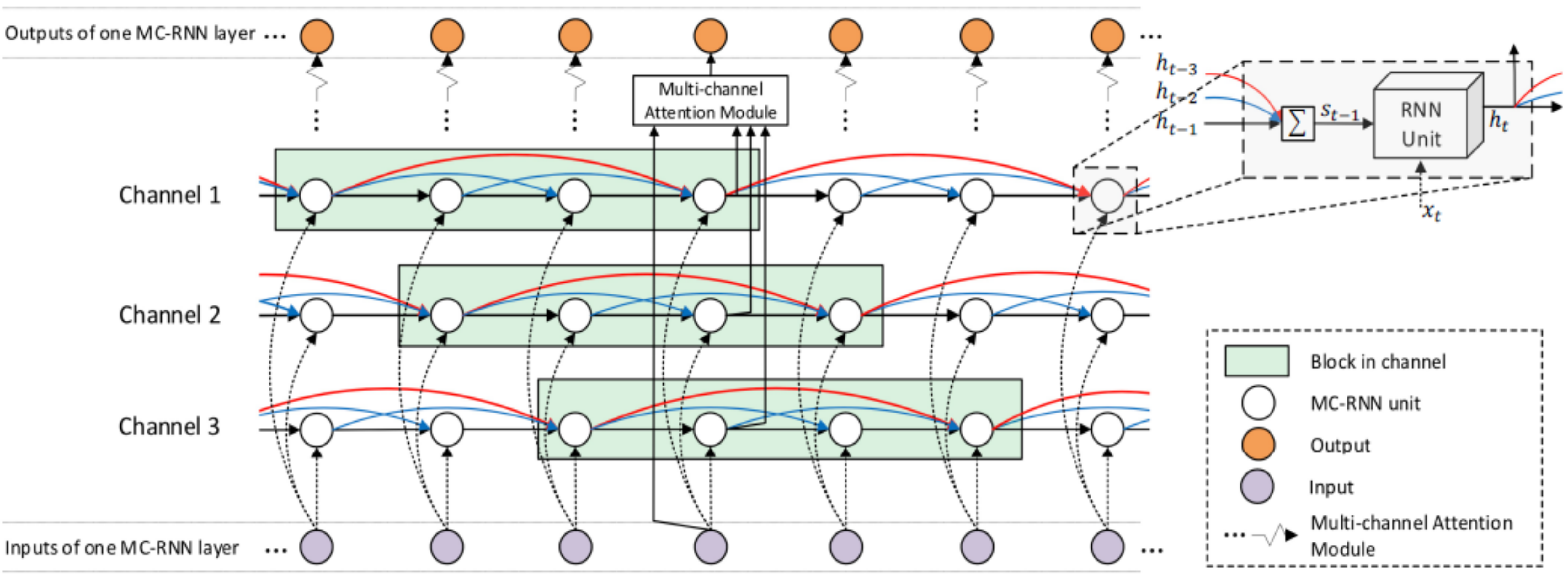


Illustration of the structure of one-layer MC-RNN with 3 channels.



- Each channel in the MC-RNN layer contains several blocks
- Local connections are built in each block
- Solid lines with the same color (red/blue/black) share the same parameter matrices
- Channels can be computed in parallel.

Capturing Rich Patterns with Multiple Channels

m_t^k denotes the number of predecessors connected to node (*t, k*).

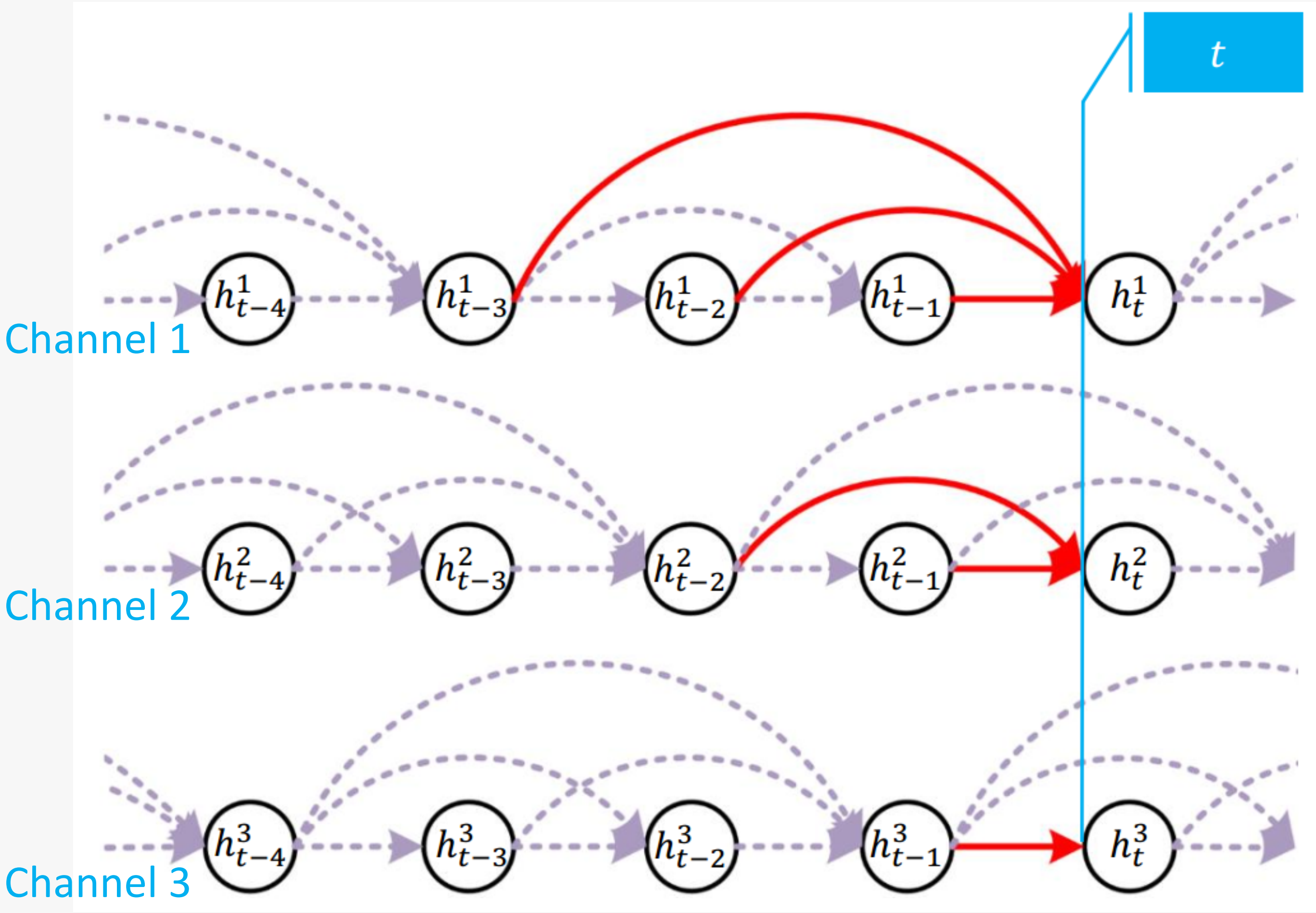
Define the temporal input at step t in channel k as

$$\boldsymbol{s}_{t-1}^{k} = \frac{1}{m_t^k} \sum_{j=1}^{m_t^k} \boldsymbol{W}_j \boldsymbol{h}_{t-j}^{k}.$$

Then apply the recurrent computation f to get the output:

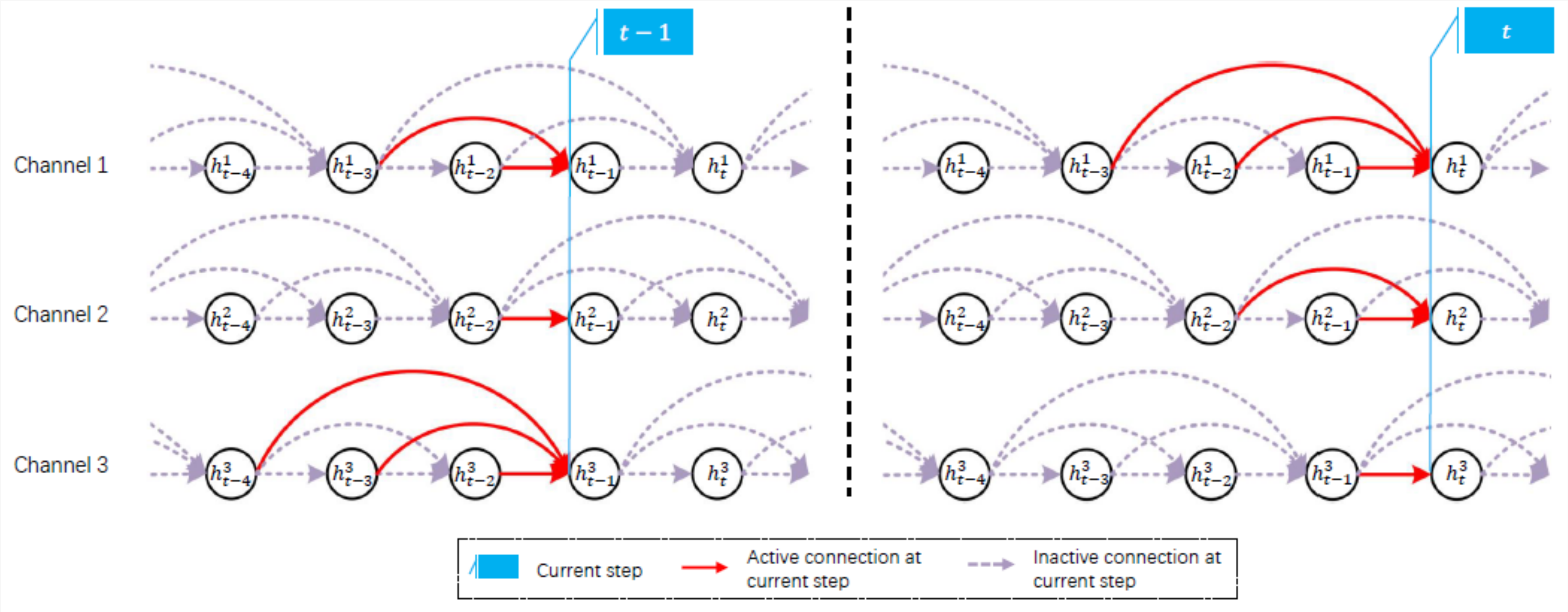
$$\boldsymbol{h}_t^k = f(\boldsymbol{s}_{t-1}^k, \boldsymbol{x}_t)$$

Learnable parameters including RNN internal parameters and weights in blocks are *shared* among different channels.



The indegree m_t^2 of Channle2 at time t in this figure is 2

- The inputs of each recurrent unit include
  - not only its immediate predecessor
  - but also from the historical units within a certain distance.
- MC-RNN can capture a strong dependence between words in a phrase, and make compact representations for the phrased
- Different Connection Mechanism for Different Channels
  - Set the blocks of neighboring channels has one step staggered with each other in a progressive way
  - All possible local structures or dependency patterns whose length is no more than the block size can be enumerated


$t-1$
t
Channel 1
Channel 2
Channel 3
h^1_{t-4} h^1_{t-3} h^1_{t-2} h^1_{t-1} h^1_t
h^2_{t-4} h^2_{t-3} h^2_{t-2} h^2_{t-1} h^2_t
h^3_{t-4} h^3_{t-3} h^3_{t-2} h^3_{t-1} h^3_t
Current step
Active connection at current step
Inactive connection at current step

At time step t, the red lines in channel 1, 2, 3 represent 4-word/ 3-word/ 2-word dependence patterns respectively

Aggregating Patterns by an Attention Module

- Combining Channels by Dynamically Adjusting Weights
  - MC-RNN is designed to have different topological connections representing different dependence patterns.
  - We use the attention mechanism to obtain the weighted average of each channel's hidden as the input to next layer, which is denoted as

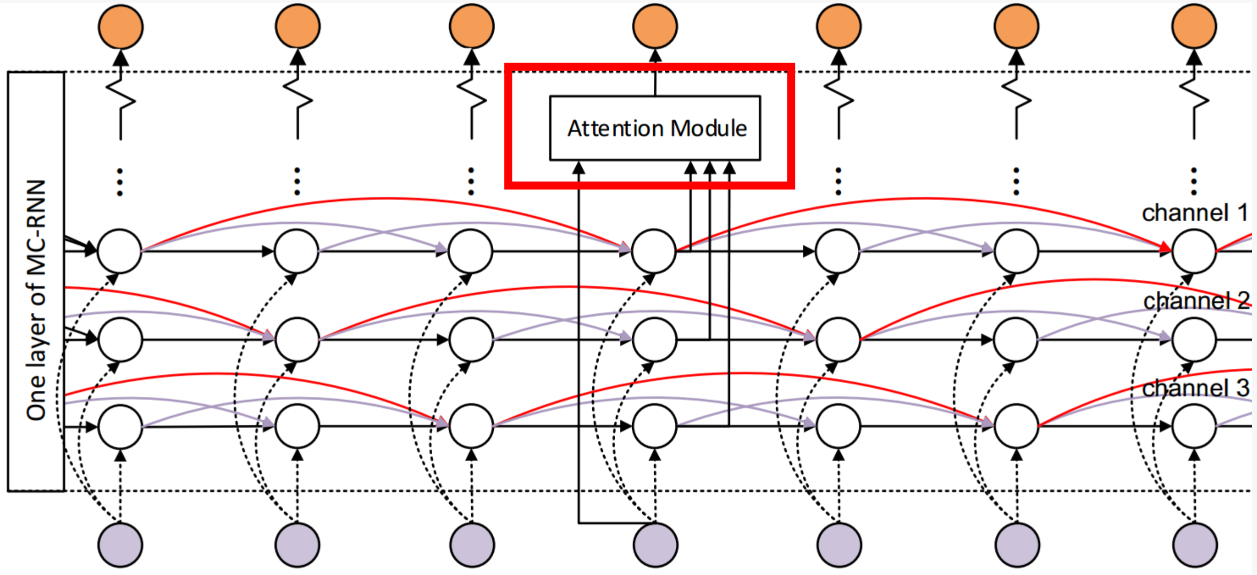
$$h_t^{att} = \sum_{k=1}^{n} \alpha_t^k h_t^k$$

  - The attention weight is calculated by

$$\alpha_t^k = \frac{\exp(e_t^k)}{\sum_{i=1}^{n} \exp(e_t^i)}$$

  - e_t^k is defined as

$$e_t^k = \boldsymbol{r}^T \tanh(\mathbf{V} \cdot \begin{bmatrix} \boldsymbol{h}_t^k \\ \boldsymbol{x}_t \end{bmatrix})$$



Content

- Motivation
  - Background of Modeling Natural Sentences
  - Challenges in Capturing Semantic Structure Information
- Our Method: Multi-Channel Recurrent Neural Networks
  - Capturing Rich Patterns with Multiple Channels
  - Aggregating Patterns by an Attention Module
- Experimental Results
- Analysis
  - Case Study and Visualization
  - Performance on Long Sentences
  - Impact of Model Size
- Conclusion

Experimental Results

- Machine Translation
  - 2-layer encoder, 2-layer decoder
  - 256-d *bpe* embedding, 256-d hidden size
  - Beam search with width 5
  - Test on IWLST 2014 De-En task

Methods	Params	BLEU
Actor-critic	-	28.53
NPMT-LM	-	29.16
HM-RNN	25M	30.60
HO-RNN	30M	31.29
Baseline-RNN	25M	31.03
MC-RNN-2	28M	31.98
MC-RNN-3	29M	**32.23**
MC-RNN-4	31M	32.09

  - Compared with
    - Baseline-RNN: the most widely used sequence to sequence framework RNNSearch (Bahdanau, Cho, and Bengio 2015)
    - HO-RNN: changed the topological structure of RNN (Soltani and Jiang 2016)
    - HM-RNN: modifies the recurrent computations (Chung, Ahn, and Bengio 2017)
    - Actor-critic: an approach to training neural networks to generate sequences using reinforcement learning (Bahdanau et al. 2017)
    - NPMT-LM: a neural phrasebased machine translation system that models phrase structures in the target language (Huang et al. 2018)

Experimental Results

- Abstractive Summarization
  - The task is to generate the headline of the given article
  - The dataset we use is Gigaword corpus (Graff et al. 2003):
    - 3.8M training article-headline pairs, 190k for validation and 2000 for test
  - MC-RNN follows the settings of Baseline-RNN:
    - Using LSTM as the recurrent unit
    - encoder and the decoder have 4 layers
    - Embedding size: 256
    - Hidden size: 256

Methods	Params	RG-1	RG-2	RG-L
HM-RNN	35M	34.68	16.11	32.22
HO-RNN	46M	35.86	16.99	33.38
Baseline-RNN	36M	34.65	16.13	32.24
MC-RNN-2	38M	36.21	17.30	33.60
MC-RNN-3	40M	**36.55**	**17.58**	**33.72**
MC-RNN-4	42M	36.50	17.44	33.68

Experimental Results

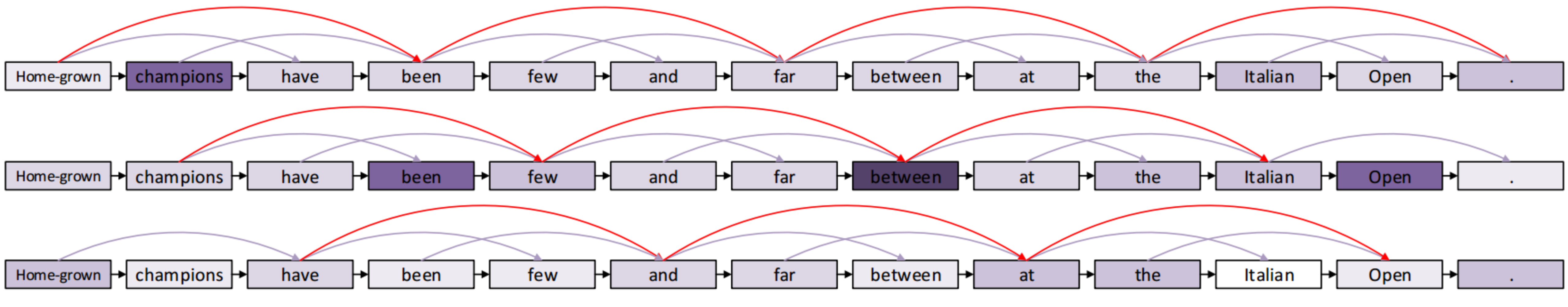
- Language Modeling
  - Evaluate on Penn Treebank corpus which contains about 1 million words
  - Evaluation metric: perplexity
  - The network structure follow the state-of-the-art model AWD-LSTM (Merity, Keskar, and Socher 2018)
    - 1150 units in the hidden layer
    - 400-d word embedding
    - DropConnect is used on the hidden-to-hidden weight matrices

Methods	Validation	Test
Variational LSTM + augmented loss (Inan, Khosravi, and Socher 2017)	71.1	68.5
Variational RHN (Zilly et al. 2016)	67.9	65.4
NAS Cell (Zoph and Le 2017)	-	62.4
Skip Connection LSTM(Melis, Dyer, and Blunsom 2018)	60.9	58.3
AWD-LSTM w/o finetune (baseline) (Merity, Keskar, and Socher 2018)	60.7	58.8
MC-RNN	**59.2**	**56.9**

Content

- Motivation
  - Background of Modeling Natural Sentences
  - Challenges in Capturing Semantic Structure Information
- Our Method: Multi-Channel Recurrent Neural Networks
  - Capturing Rich Patterns with Multiple Channels
  - Aggregating Patterns by an Attention Module
- Experimental Results
- Analysis
  - Case Study and Visualization
  - Performance on Long Sentences
  - Impact of Model Size
- Conclusion

Case Studies and Visualization

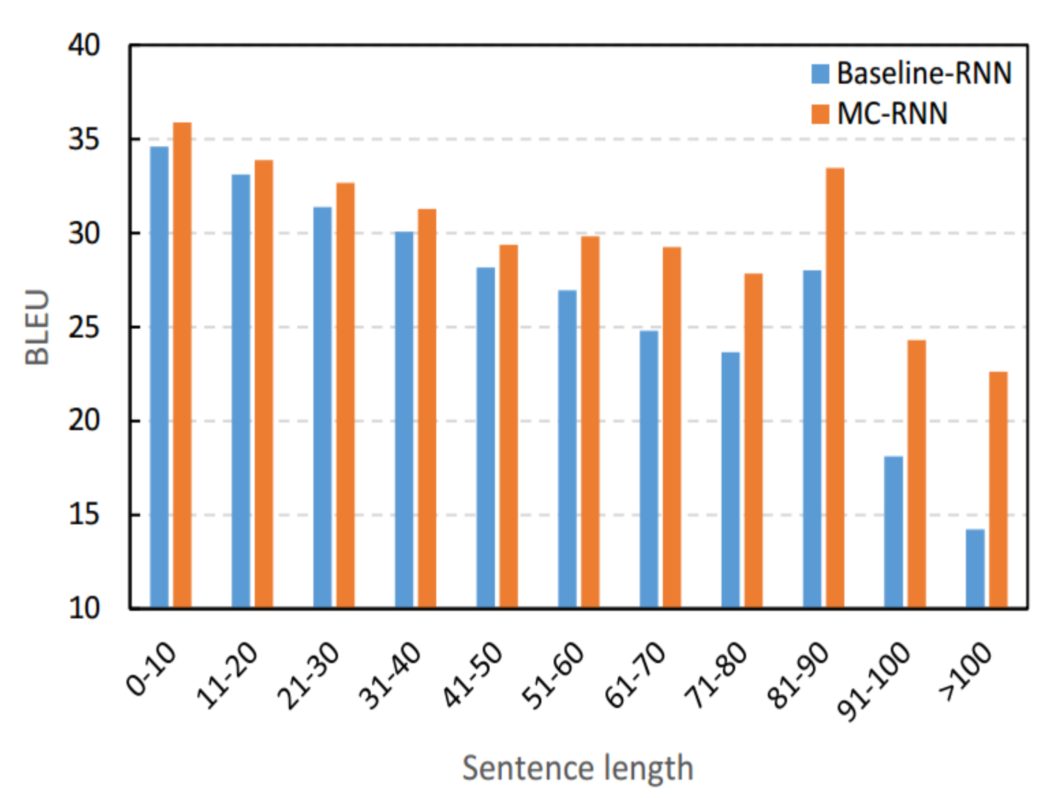


Visualization of attention scores of the sentence “ Home-grown champions have been few and far between at the Italian Open.”

- Local dependence patterns and local structures are captured, such as:
  - ”home-grown champions”
  - ”champions have been”
  - ”few and far between”
  - ”Italian Open”

Performance on Long Sentences

- Conducted on IWSLT-14 De-En translation task
- Long sentences are more difficult to handle than short ones
  - Both our method and the baseline-RNN model perform worse as the lengths of the sentences increase, indicating
- Our model brings much more improvement on long sentences
  - when the sentence length is greater than 61, our model outperforms baselines by a larger margin
- **MC-RNN enables short-cut connections across timestep and directly passes error signal through blocks**



Impact of Model Size and Time Cost

- We tried several runs for Baseline-RNN
  - **Baseline-RNN-large**: increase the size of the hidden state from 256 to 286
  - **Baseline-RNN-deep**: Increase the number of layers from 2 to 3
- No significant improvement of performance on Baseline-RNN
- Better performance of our MC-RNN is caused by model design rather than larger model size
- Owing to parallel computation, MC-RNN can achieve almost the same time cost as the conventional RNN

Methods	Params	BLEU
Baseline-RNN	25M	31.03
Baseline-RNN-large	29M	30.93
Baseline-RNN-deep	29M	30.98
MC-RNN-2	28M	31.98
MC-RNN-3	29M	**32.23**
MC-RNN-4	31M	32.09

Content

- Motivation
  - Background of Modeling Natural Sentences
  - Challenges in Capturing Semantic Structure Information
- Our Method: Multi-Channel Recurrent Neural Networks
  - Capturing Rich Patterns with Multiple Channels
  - Aggregating Patterns by an Attention Module
- Experimental Results
- Analysis
  - Case Study and Visualization
  - Performance on Long Sentences
  - Impact of Model Size
- Conclusion

Conclusions

- We proposed a new RNN model with multichannel multi-block structure to *better capture and utilize local patterns in sequential data* for language-related tasks

- Experiments on machine translation, abstractive summarization, and language modeling validated the effectiveness of the proposed model
  - Achieved new state-of-the-art results on Gigaword on text summarization and Penn Treebank on language modeling

Thanks!

Contact info

Chang Xu
Email：changxu@nbjl.nankai.edu.cn
Homepage： https://chang-xu.github.io
5-th year student of Joint Ph.D Program with Microsoft Research Asia