Paper Reading

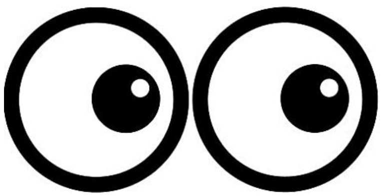
Learning to Combine Top-Down and Bottom-Up Signals in Recurrent Neural Networks with Attention over Modules (ICML2020)

Jianjie Luo

2020.12.18

In real life

What is Bottom-up(BU) Signal?

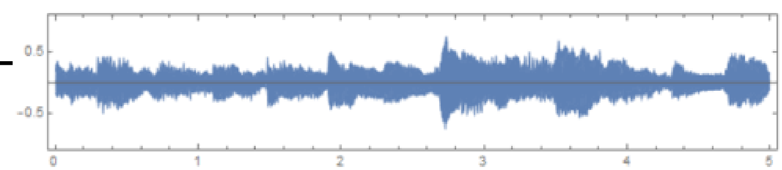


Watch

Read

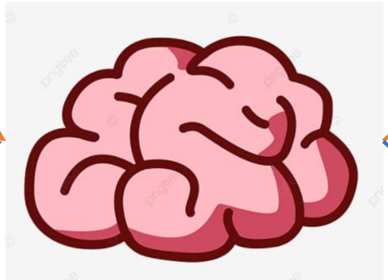
He wakes up. He sees the sun rise. He brushes his teeth. His teeth are white. He puts on his clothes. His shirt is blue. His shoes are yellow. His pants are brown. He goes downstairs. He gets a bowl. He pours milk and cereal. He eats. He gets the newspaper. He reads.

Listen



What is Top-down(TD) Signal?

“西红柿炒____”

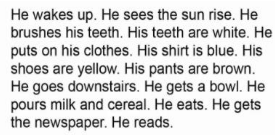


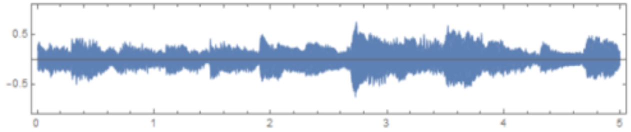
“鸡蛋”

“Blue Car”

BU/TD Signals help us make Decision/Prediction

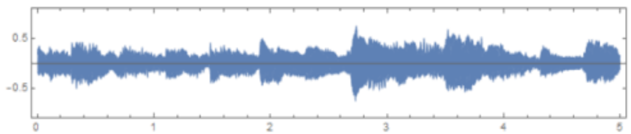
Why we need both BU&TD signals better?

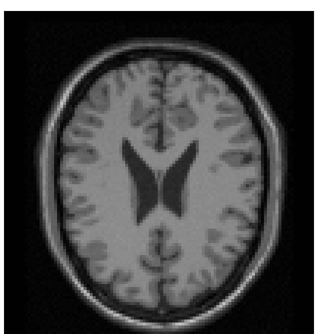

He wakes up. He sees the sun rise. He brushes his teeth. His teeth are white. He puts on his clothes. His shirt is blue. His shoes are yellow. His pants are brown. He goes downstairs. He gets a bowl. He pours milk and cereal. He eats. He gets the newspaper. He reads.



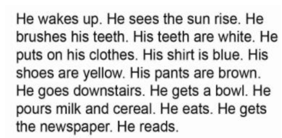
Why we need both BU&TD signals better?

He wakes up. He sees the sun rise. He brushes his teeth. His teeth are white. He puts on his clothes. His shirt is blue. His shoes are yellow. His pants are brown. He goes downstairs. He gets a bowl. He pours milk and cereal. He eats. He gets the newspaper. He reads.

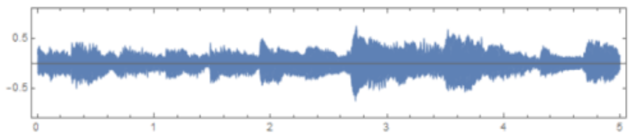




Why we need both BU&TD signals better?



He wakes up. He sees the sun rise. He brushes his teeth. His teeth are white. He puts on his clothes. His shirt is blue. His shoes are yellow. His pants are brown. He goes downstairs. He gets a bowl. He pours milk and cereal. He eats. He gets the newspaper. He reads.



Provide precise info

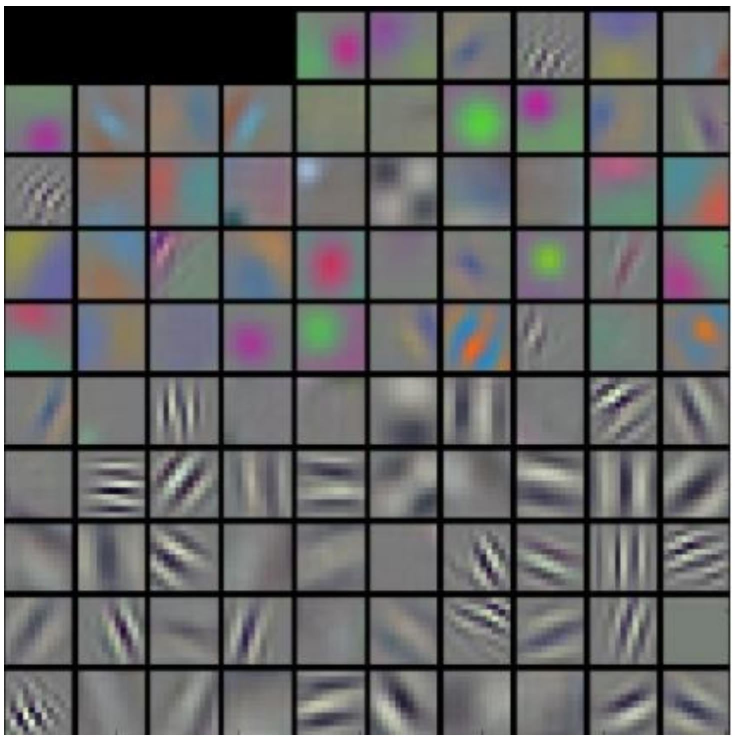
Complementary

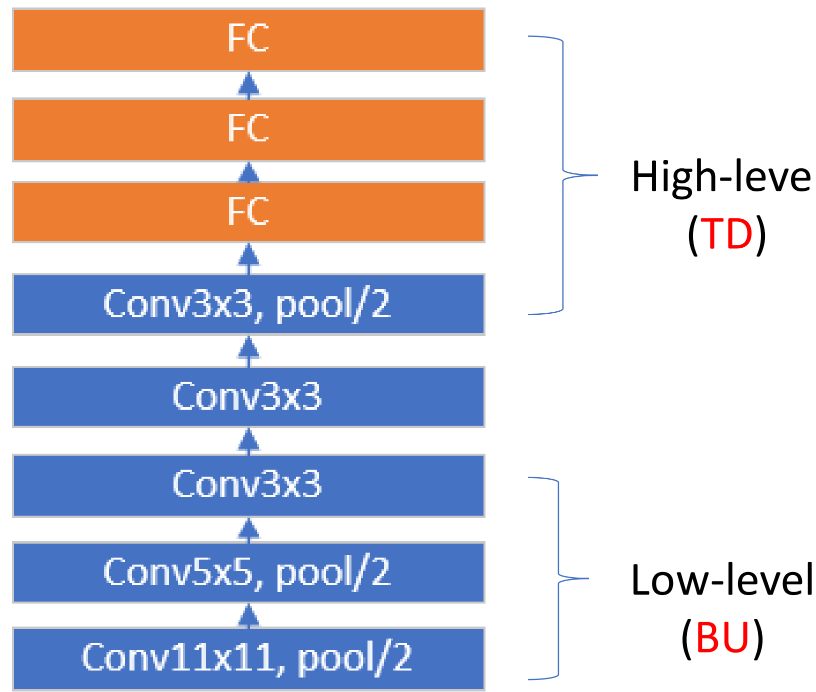
Filter noisy data

Making Robust Decision/Prediction

From real life to Neural Networks

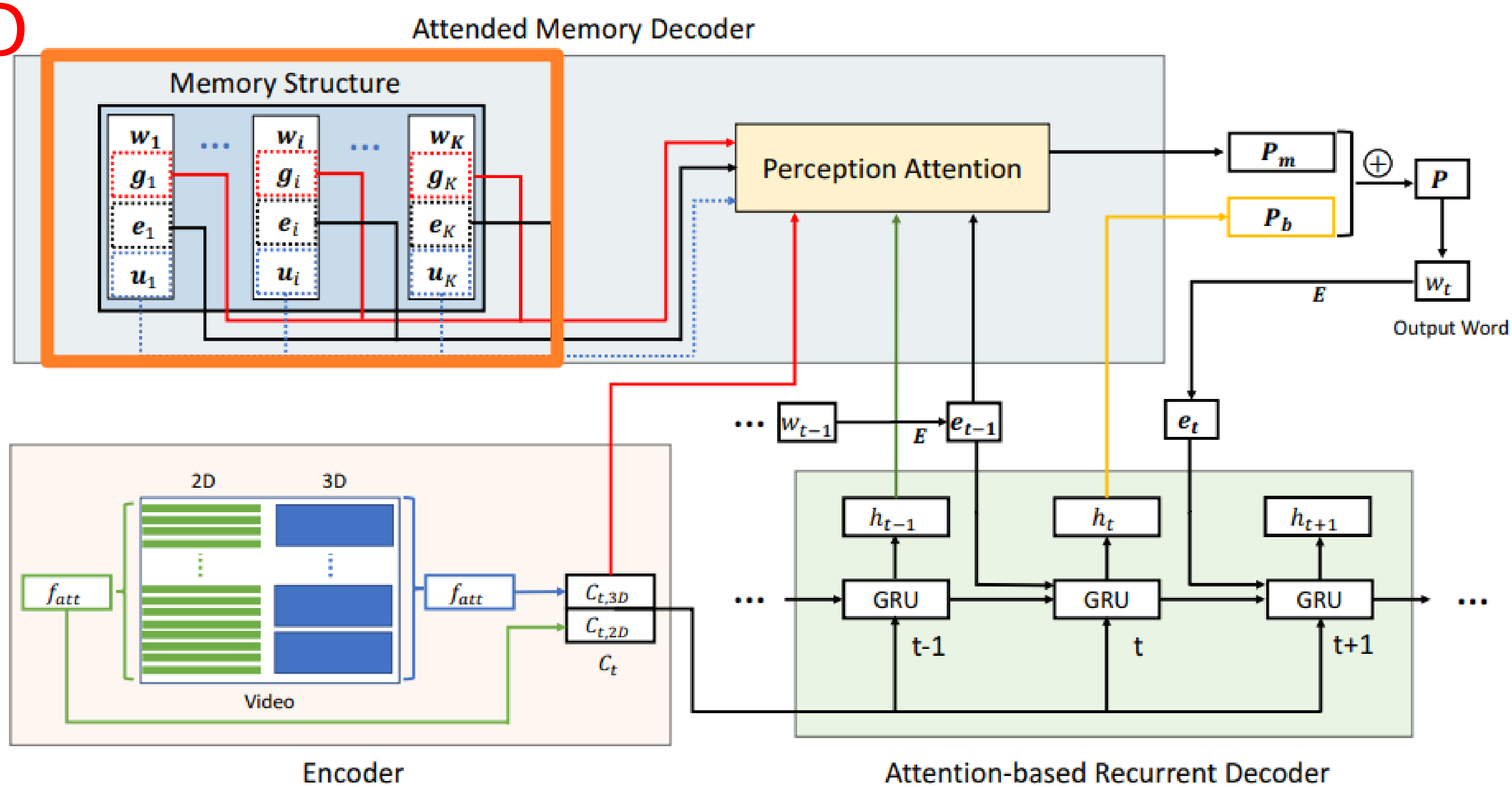
Where are the BU&TD signals in Neural Networks?



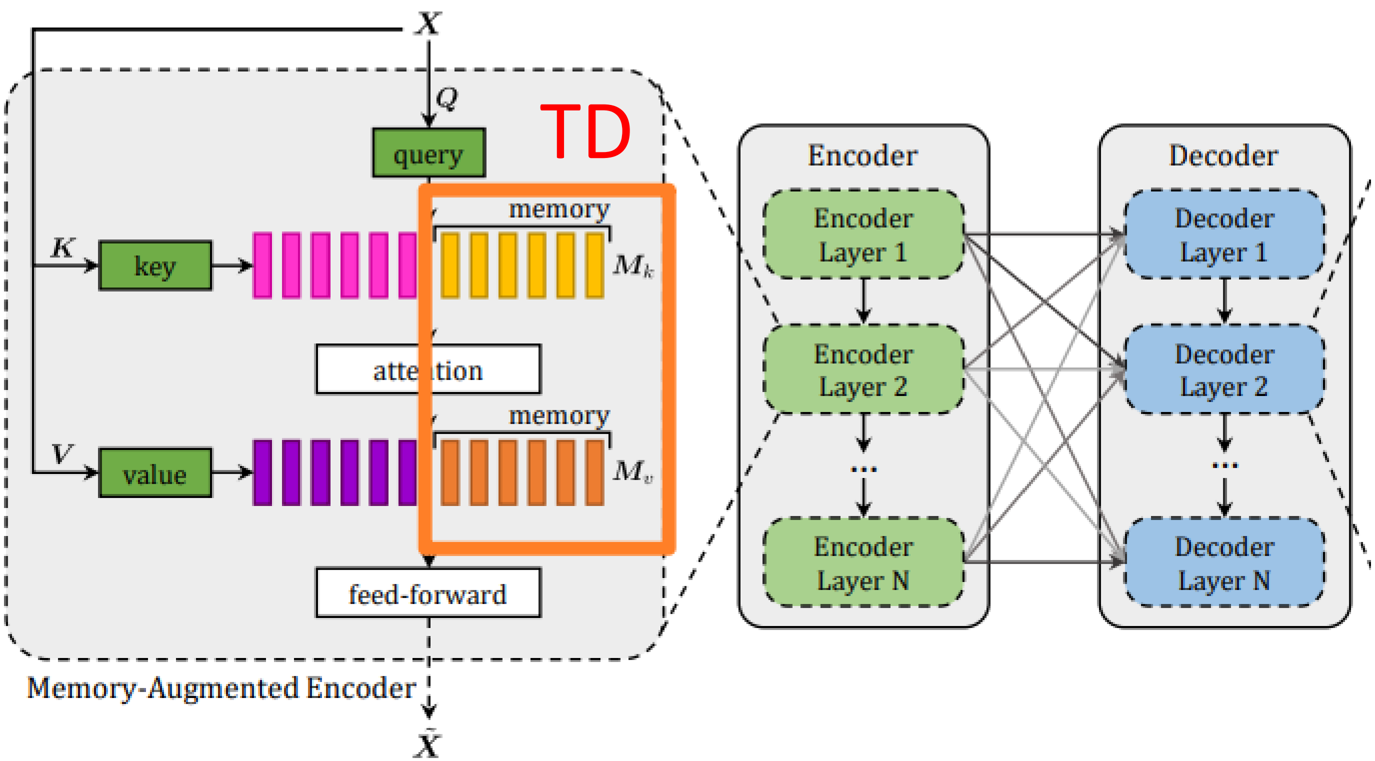


Where are the BU&TD signals in Neural Networks?

TD



Where are the BU&TD signals in Neural Networks?



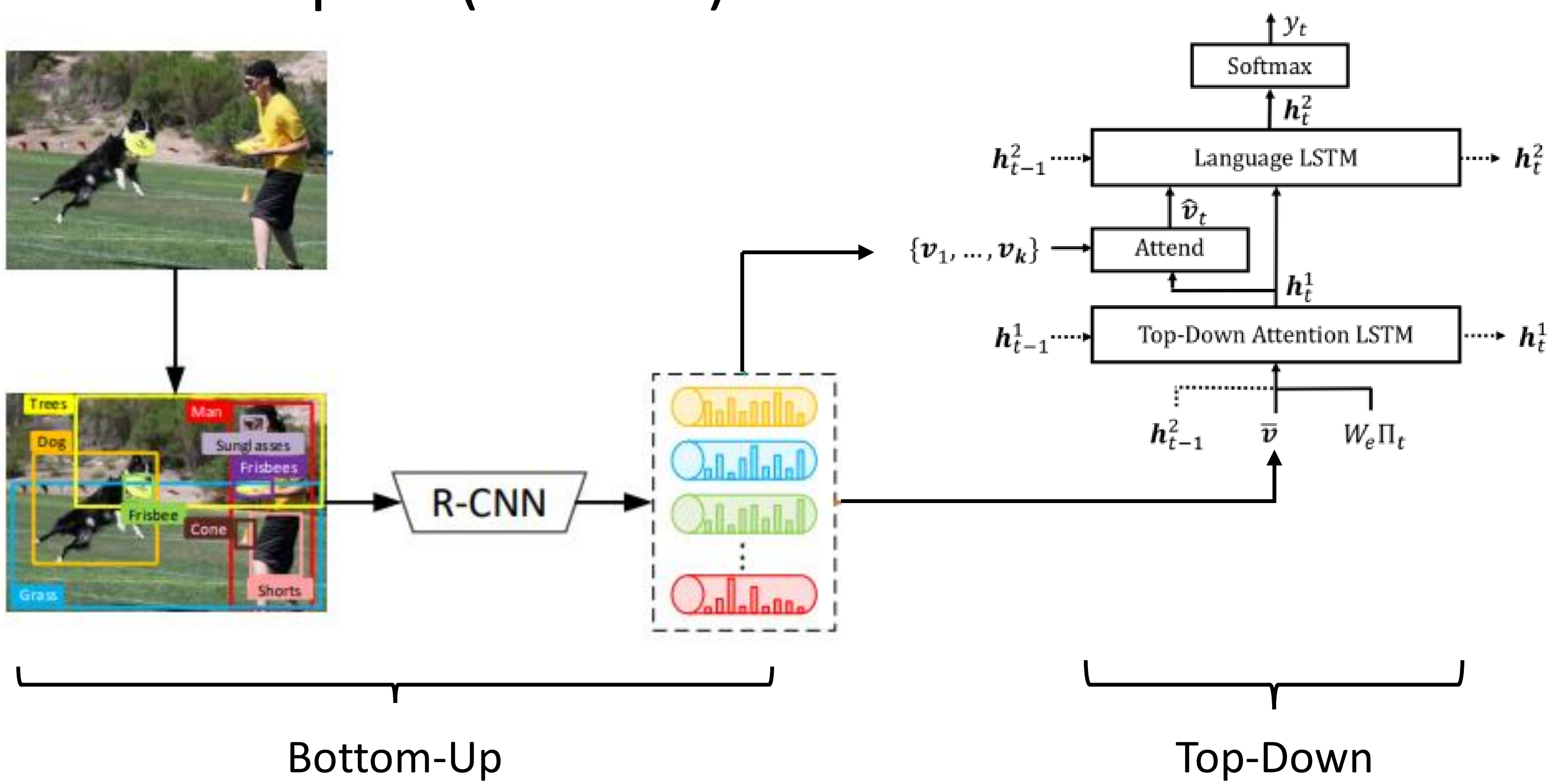
How to Combine BU&TD (in general)?

Every Fusion Methods you can imagine…

- Add Fusion
- Mean Fusion
- Hadamard product Fusion
- Concat Fusion
- Attention Fusion ★
- …

How to Combine BU&TD (in RNN)?

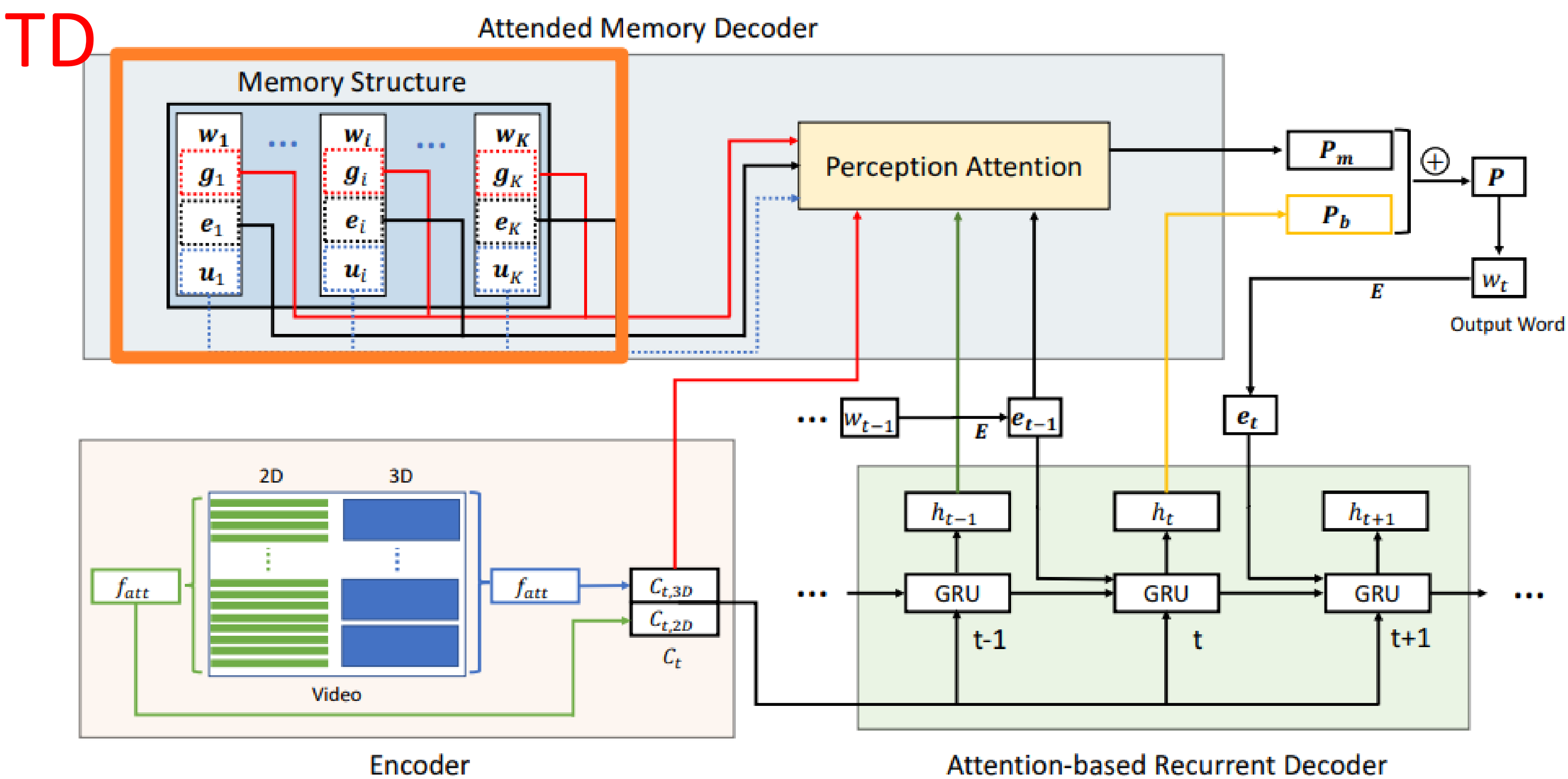
- Other Papers (BUTD[1])



[1] *Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. Peter Anderson et al. In CVPR2018*

How to Combine BU&TD (in RNN)?

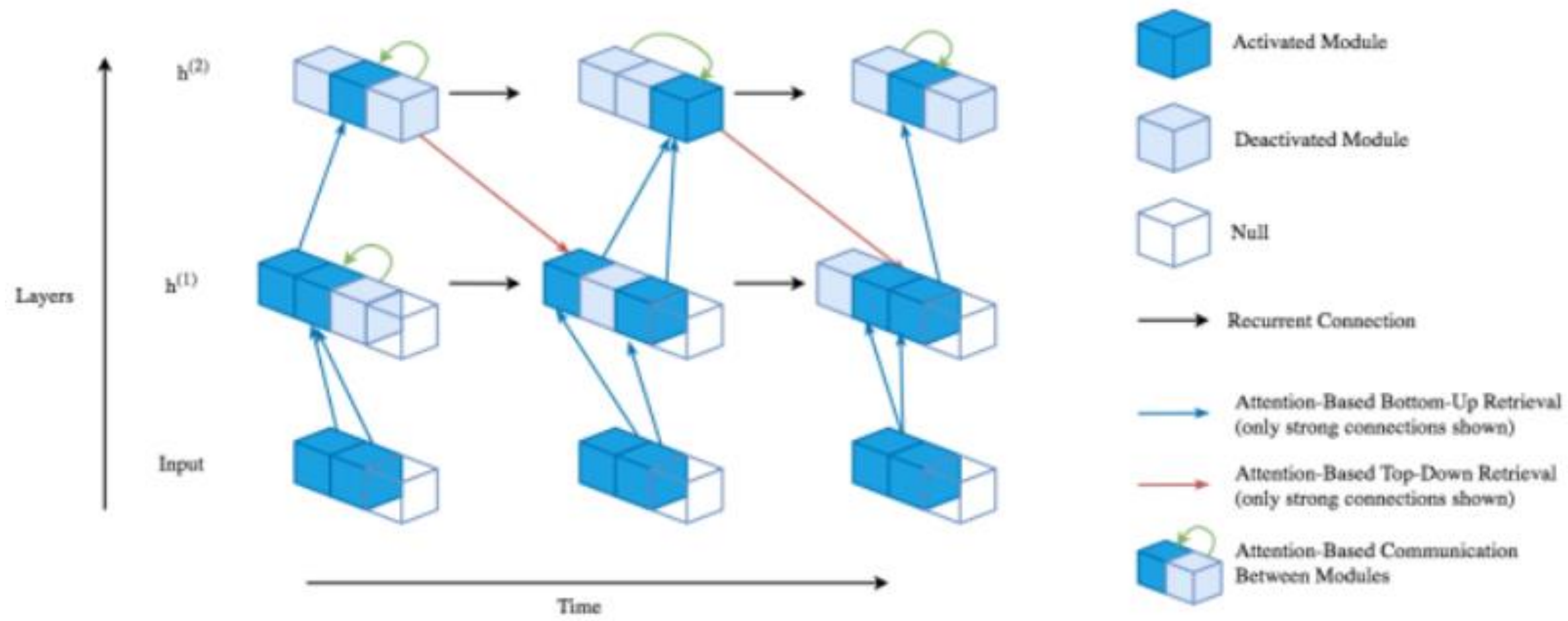
- Other Papers (MARN[1])



[1] Memory-Attended Recurrent Network for Video Captioning. Wenjie Pei et al. In CVPR2019

How to Combine BU&TD (in RNN)?

- This Paper (BRIMs[1])



[1] Learning to Combine Top-Down and Bottom-Up Signals in Recurrent Neural Networks with Attention over Modules. S Mittal et al. In ICML2020

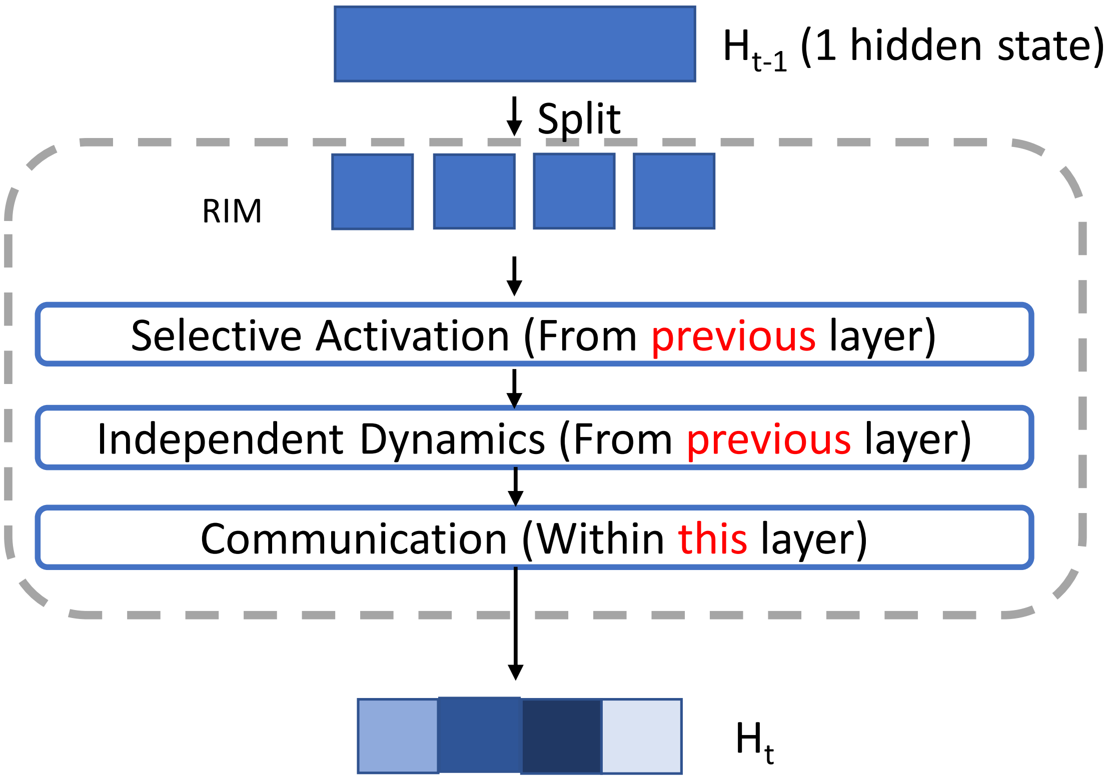
Technical Details (BRIMs)

Preliminaries for BRIMs

- Multi-layer stacked RNN
- Key-Value Attention
- RIMs
  - Selective Activation
  - Independent Dynamics
  - Communication

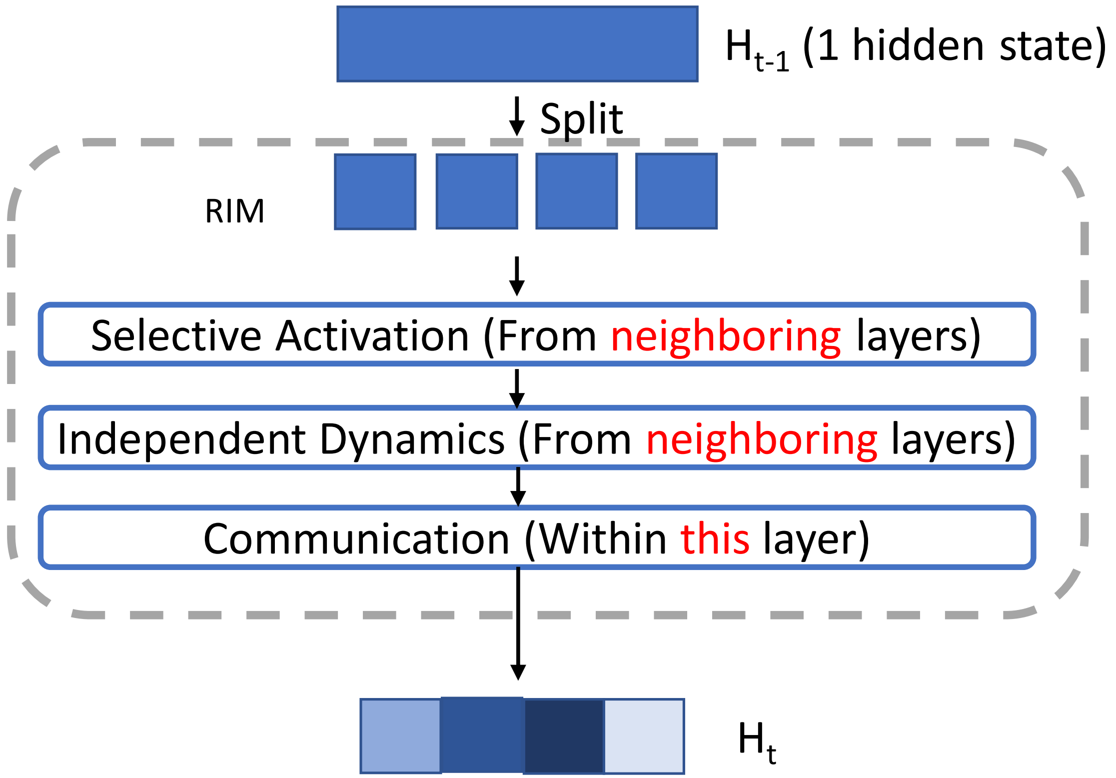
Technical Details (BRIMs)

- Recurrent Independent Mechanisms(RIMs)



Technical Details (BRIMs)

- Bidirectional Recurrent Independent Mechanisms(BRIMs)



Technical Details (BRIMs)

- Pseudo-Code for BRIMs

Algorithm 1: Single recurrent step for an L layered BRIMs model

Result: RNN Cell forward for L layered BRIMs

x : Input

h_l : Hidden state of layer l represented as flat vector

$h_l[k]$: Hidden state of k^{th} module of layer l

n_l : Number of modules in layer l

m_l : Number of modules kept active in layer l

ϕ : Null vector

All `Query`, `Key`, `Value` networks are fully connected neural networks

$\mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{A_S}, \mathbf{A_R}$ denote matrices

Note: Unless specified, all indexing start with 1

Technical Details (BRIMs)

- Pseudo-Code for BRIMs

```
Function BRIMsCell (x, h_1, ..., h_{n_l}) :
    h_0 = x

    for l = 1 to L do
        for k = 1 to n_l do
            Q[k] = Input Query_{l,k}(h_l[k])
        end
        K[0], V[0] = Null Key Value_l(φ)
        K[1], V[1] = Input Key Value_l(h_{l-1})
        K[2], V[2] = Top-Down Key Value_l(h_{l+1})      (if available)
        A_S = Softmax(QK^T / sqrt(d_att) )
        A_R = A_S V

        Sort A_S[:,0] and take lowest m_l as active
        for k s.t. module k is active do
            h_l[k] = RNN_{l,k}(A_R[k], h_l[k])            (can use GRU or LSTM)
        end

        for k = 1 to n_l do
            Q[k] = Communication Query_{l,k}(h_l[k])
            K[k] = Communication Key_{l,k}(h_l[k])
            V[k] = Communication Value_{l,k}(h_l[k])
        end
        A_R = Softmax(QK^T / sqrt(d_att)) V
        for k s.t. module k is active do
            h_l[k] += A_R[k]
        end

    end
    return h
```

Selective Activation

Independent Dynamics

Communication within layer

Differences between BRIMs and other network designs

	H	A	B	M
LSTM	Optional	Optional	Optional	×
RMC[1]	×	√	×	×
Transformer	√	√	√	×
RIMs	Optional	√	×	√
BRIMs	√	√	√	√

Properties Notions:

H: Hierarchy

A: Attention

B: Bidirectional Information Flow

M: Modularity

[1] Relational recurrent neural networks. Santoro, A et al. In Arxiv2018

How to Apply BRIMs?

ONLY focus on Network(RNN) Design.

Do ANYTHING the RNNs can!

Experiments

Task Lists

- Sequential MNIST and CIFAR ★
- Adding tasks
- Moving MNIST ★
- Handling Occlusion in Bouncing Balls
- Language Modeling
- Reinforcement Learning

Sequential MNIST/CIFAR

- **Dataset**: MNIST / CIFAR
- **Task descriptions**: a classification task with a sequence length of T = 784 or 1024
- **Input**: One pixel at a time
- **Output**: 1 label

Sequential MNIST

- Results

Algorithm	*Properties*	*16×16*	*19×19*	*24×24*
LSTM	—	86.8	42.3	25.2
LSTM	H	87.2	43.5	22.9
LSTM	H+B	83.2	44.4	25.3
LSTM	H+A	84.3	47.5	31.0
LSTM	H+A+B	83.2	40.1	20.8
RMC	A	89.6	54.2	27.8
Transformers	H+A+B	91.2	51.6	22.9
RIMs	A+M	88.9	67.1	38.1
Hierarchical RIMs	H+A+M	85.4	72.0	50.3
MLD-RIMs	H+A+M	88.8	69.1	45.3
BRIMs (ours)	H+A+B+M	88.6	74.2	51.4

Table 1. Performance on the **Sequential MNIST resolution generalization:** Test Accuracy % after 100 epochs. All models were trained on 14x14 resolution but evaluated at different resolutions; results averaged over 3 different trials.

Sequential CIFAR

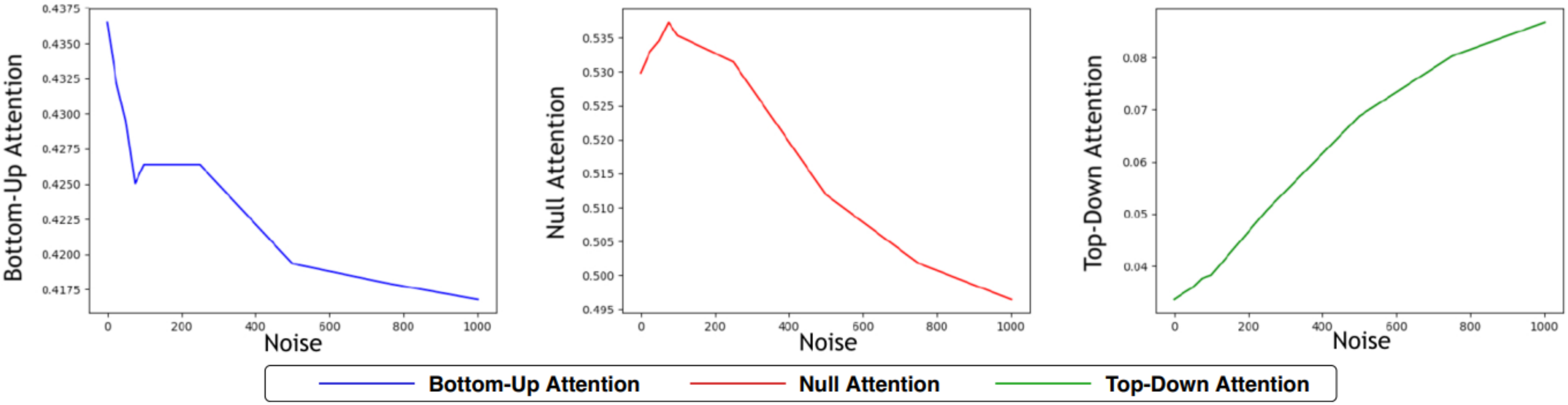
- Results

Algorithm	*Properties*	*19×19*	*24×24*	*32×32*
LSTM	—	54.4	44.0	32.2
LSTM	H	57.0	46.8	33.2
LSTM	H+B	56.5	52.2	42.1
LSTM	H+A	56.7	51.5	40.0
LSTM	H+A+B	59.9	54.6	43.0
RMC	A	49.9	44.3	31.3
RIMs	A+M	56.9	51.4	40.1
Hierarchical RIMs	H+A+M	57.2	54.6	46.8
MLD-RIMs	H+A+M	56.8	53.1	44.5
BRIMs (ours)	H+A+B+M	60.1	57.7	52.2

Table 2. Performance on **Sequential CIFAR generalization:** Test Accuracy % after 100 epochs. Both the proposed and the Baseline model (LSTM) were trained on 16x16 resolution but evaluated at different resolutions; results averaged over 3 different trials.

Sequential CIFAR

- Analysis of Top Down Attention

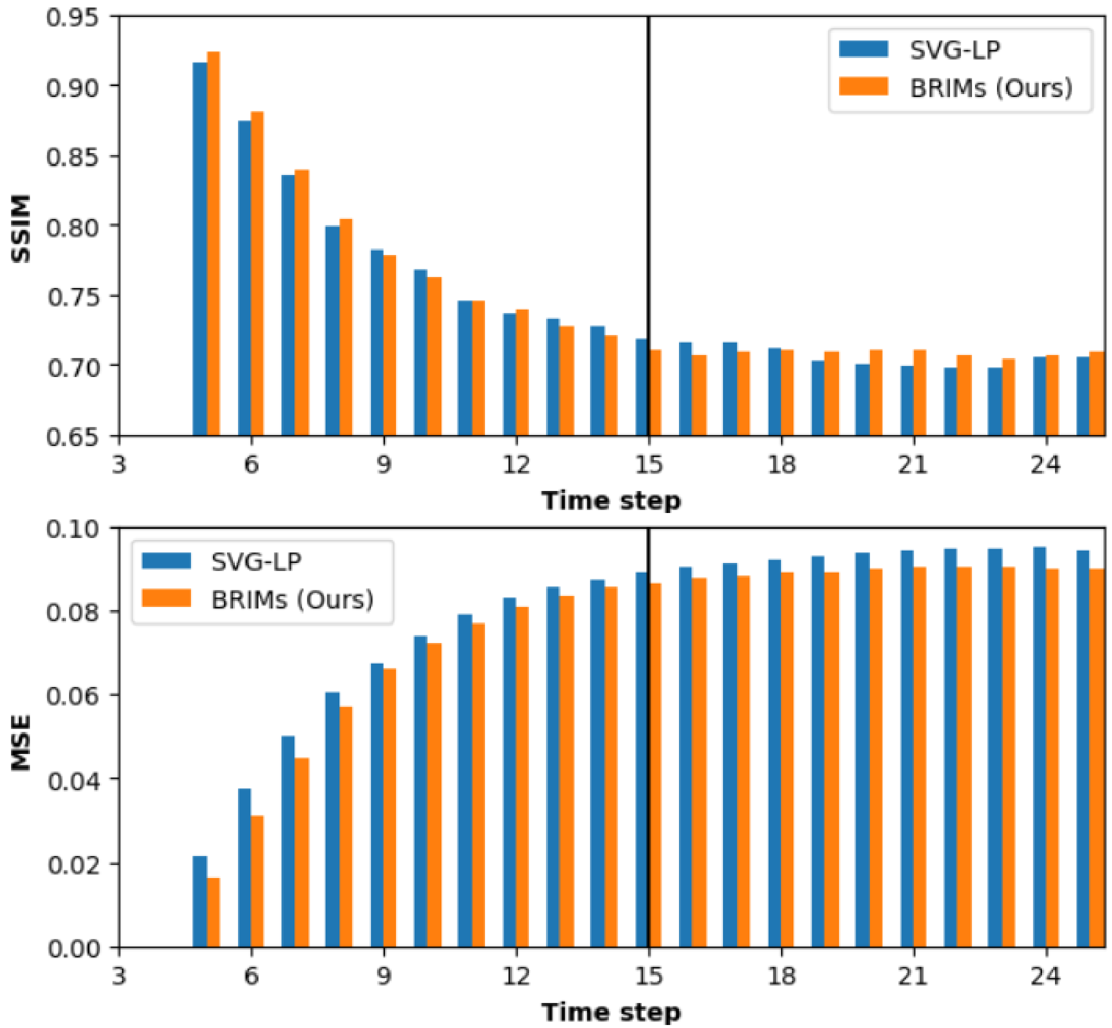


Moving MNIST

- **Dataset**: Moving MNIST 
- **Task descriptions**: a video(seq) prediction task
- **Input**: 5 frames sampled on the fly from a video
- **Output**: predict the next 10 frames in the sequence

Moving MNIST

- Results



Conclusion

- Top-down and bottom-up information are both critical to robust and accurate perception

- The combination method used in BRIMs can get a more robust RNN model

Two Key points in BRIMs

1. The modular decomposition in hidden state features

2. A new connection from High-level information to Low-level information, i.e. Top-down connection

Thanks