DL4NLP: Challenges and Future Directions

Xipeng Qiu
xpqiu@fudan.edu.cn
http://nlp.fudan.edu.cn/~xpqiu

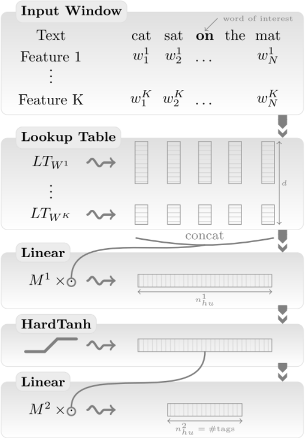
Fudan University

November 14, 2015
CCL 2015
Guangzhou, China

Outline

1. Neural Models for NLP
2. DL4NLP at Fudan NLP Lab
   - Our Focused Problem: Feature Composition
   - Convolutional Neural Tensor Network
   - Recursive Neural Network for Dependency Parse Tree
   - Gated Recursive Neural Network
   - Multi-Timescale LSTM
3. Future Directions
   - Memory Mechanism
   - Attention Mechanism
   - Novel Applications

General Neural Architectures for NLP



1. represent the words/features with dense vectors (embeddings) by lookup table;
2. concatenate the vectors;
3. classify/match/rank with multi-layer neural networks.

from [Collobert et al., 2011]

Difference with the traditional methods

	Traditional methods	Neural methods
Features	Discrete Vector (One-hot Representation)	Dense Vector (Distributed Representation)
	High-dimension	Low-dimension
Classifier	Linear	Non-Linear

General Neural Architectures for NLP

- **Word Level**
  - NNLM
  - C&W
  - CBOW & Skip-Gram
- **Sentence Level**
  - **NBOW**
  - **Sequence Models**: Recurrent NN, LSTM, Paragraph Vector
  - **Topoligical Models**: Recursive NN,
  - **Convolutional Models**: DCNN
- **Document Level**
  - **NBOW**
  - **Hierachical Models** two-level CNN
  - **Sequence Models** LSTM, Paragraph Vector

Our Focused Problem: Feature Composition

Not "Really" Deep Learning in NLP

- Most of the neural models is very shallow in NLP.
- The major benefit is introducing dense representation.
- The feature composition is also quite simple.
  - Concatenation
  - Sum/Average
  - Bilinear model

Quite Simple Feature Composition

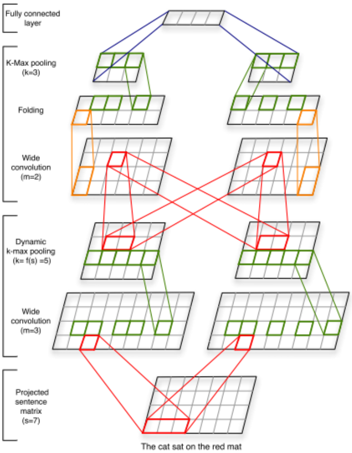
Given two embeddings **a** and **b**,

1. how to calculate their similarity/relevence/relation?
   1. Concatenation
      $$\mathbf{a} \oplus \mathbf{b} \rightarrow \text{ANN} \rightarrow \text{output}$$
   2. Bilinear
      $$\mathbf{a}^T \mathbf{M} \mathbf{b} \rightarrow \text{output}$$
2. how to use them in classification task?
   1. Concatenation
      $$\mathbf{a} \oplus \mathbf{b} \rightarrow \text{ANN} \rightarrow \text{output}$$
   2. Sum/Average
      $$\mathbf{a} + \mathbf{b} \rightarrow \text{ANN} \rightarrow \text{output}$$

Problem

How to enhance the neural model without increasing the network depth?

Convolutional Neural Network (CNN)



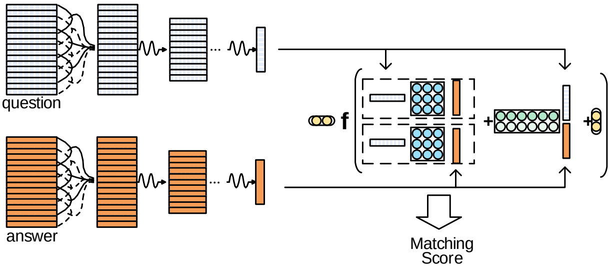
Key steps

- Convolution
- (optional) Folding
- Pooling

Various models

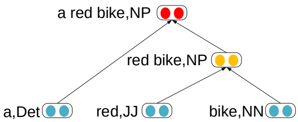
- DCNN (k-max pooling) [Kalchbrenner et al., 2014]
- CNN (binary pooling) [Hu et al., 2014]
- ...

Convolutional Neural Tensor Network for Text Matching [Qiu and Huang, 2015]



Architecture of Convolutional Neural Tensor Network

Recursive Neural Network (RecNN) [Socher et al., 2013]



Topological models compose the sentence representation following a given topological structure over the words.

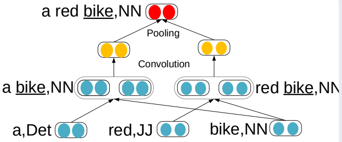
Given a labeled binary parse tree, $((p_2 \rightarrow ap_1), (p_1 \rightarrow bc))$, the node representations are computed by

$$\mathbf{p}_1 = f(\mathbf{W} \begin{bmatrix} \mathbf{b} \\ \mathbf{c} \end{bmatrix}),$$

$$\mathbf{p}_2 = f(\mathbf{W} \begin{bmatrix} \mathbf{a} \\ \mathbf{p}_1 \end{bmatrix}).$$

A variant of RecNN for Dependency Parse Tree [Zhu et al., 2015]

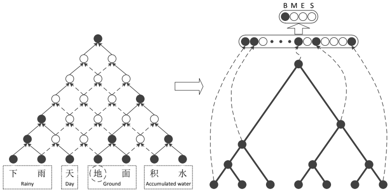
Recursive neural network can only process the binary combination and is not suitable for dependency parsing.



Recursive Convolutional Neural Network

- introducing the convolution and pooling layers;
- modeling the complicated interactions of the head word and its children.

Gated Recursive Neural Network [Chen et al., 2015a]



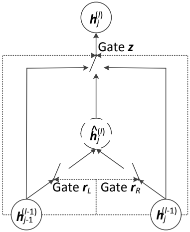
- DAG based Recursive Neural Network
- Gating mechanism

An relative complicated solution

GRNN models the complicated combinations of the features, which selects and preserves the useful combinations via reset and update gates.

A similar model: AdaSent [Zhao et al., 2015]

GRNN Unit

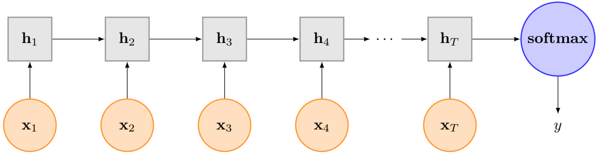


Two Gates

- reset gate
- update gate

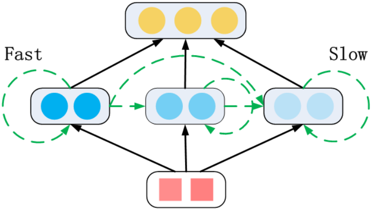
- Chinese Word Segmentation [Chen et al., 2015a]
- Dependency Parsing [Chen et al., 2015c]
- Sentence Modeling [Chen et al., 2015b]

Unfolded LSTM for Text Classification



Drawback: long-term dependencies need to be transmitted one-by-one along the sequence.

Multi-Timescale LSTM

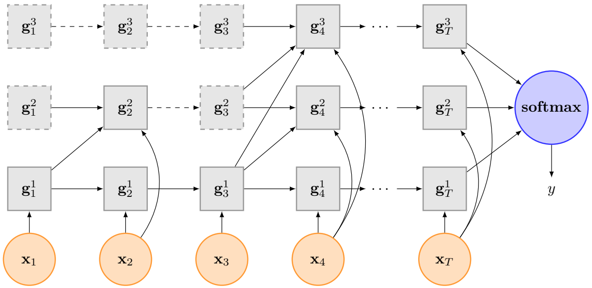


(a) Fast-to-Slow Strategy (b) Slow-to-Fast Strategy

Figure: Two feedback strategies of our model. The dashed line shows the feedback connection, and the solid link shows the connection at current time.

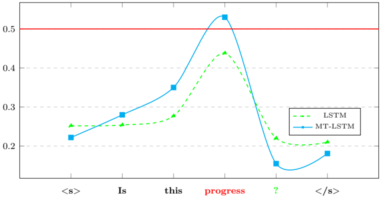
from [Liu et al., 2015]

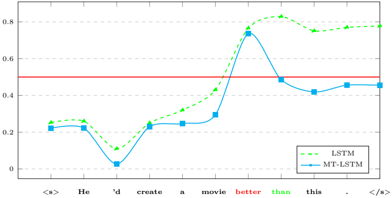
Unfolded Multi-Timescale LSTM with Fast-to-Slow Feedback Strategy

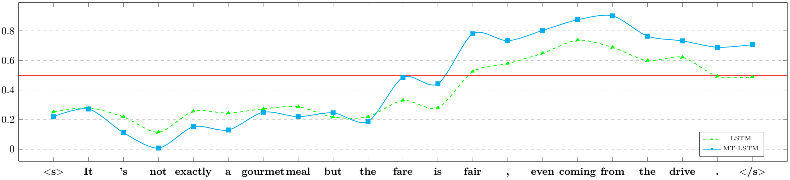


from [Liu et al., 2015]

LSTM for Sentiment Analysis







Memory Mechanism

What differences among the various models from memory view?

	Short-term	long-term	Global	External
SRN	Yes	No	No	No
LSTM/GRU	Yes	No	Maybe	No
PV	Yes	Yes	Yes	No
NTM/DMN	Maybe	Maybe	Maybe	Yes

Attention Mechanism

Neural Models as Components

- Component models could be more complex than main model.
- More attention mechanisms?

Novel Applications

- Abstractive Summarization
- Text Generation
- Integration of Syntax, Semantics and Knowledge
- ...

References I

Xinchi Chen, Xipeng Qiu, Chenxi Zhu, and Xuanjing Huang. Gated recursive neural network for Chinese word segmentation. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, 2015a.

Xinchi Chen, Xipeng Qiu, Chenxi Zhu, Shiyu Wu, and Xuanjing Huang. Sentence modeling with gated recursive neural network. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2015b.

Xinchi Chen, Yaqian Zhou, Chenxi Zhu, Xipeng Qiu, and Xuanjing Huang. Transition-based dependency parsing using two heterogeneous gated recursive neural networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2015c.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537, 2011.

References II

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. Convolutional neural network architectures for matching natural language sentences. In *Advances in Neural Information Processing Systems*, 2014.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. In *Proceedings of ACL*, 2014.

PengFei Liu, Xipeng Qiu, Xinchi Chen, Shiyu Wu, and Xuanjing Huang. Multi-timescale long short-term memory neural network for modelling sentences and documents. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2015.

Xipeng Qiu and Xuanjing Huang. Convolutional neural tensor network architecture for community-based question answering. In *Proceedings of International Joint Conference on Artificial Intelligence*, 2015.

Richard Socher, John Bauer, Christopher D Manning, and Andrew Y Ng. Parsing with compositional vector grammars. In *In Proceedings of the ACL conference*. Citeseer, 2013.

References III

Han Zhao, Zhengdong Lu, and Pascal Poupart. Self-adaptive hierarchical sentence model. *arXiv preprint arXiv:1504.05070*, 2015.

Chenxi Zhu, Xipeng Qiu, Xinchi Chen, and Xuanjing Huang. A re-ranking model for dependency parser with recursive convolutional neural network. In *Proceedings of Annual Meeting of the Association for Computational Linguistics*, 2015.