Deep Learning for Natural Language Processing
深度学习在自然语言处理的应用

Hang Li
Noah’s Ark Lab
Huawei Technologies
Outline of Lecture

• Introduction
• Basics of DL for NLP
• State of the Art of DL for NLP
• Previous Work at Noah’s Ark Lab
• Recent Progress at Noah’s Ark Lab
• Advantages and Disadvantages
• Summary
Ultimate Goal: Natural Language Understanding

Natural Language Dialogue

Text Comprehension
Natural Language Understanding

• Two definitions:
  – Representation-based: if system creates proper internal representation, then we say it “understands” language
  – Behavior-based: if system properly follows instruction in natural language, then we say it “understands” language, e.g., “bring me a cup of tea”

• We take the latter definition
Five Characteristics of Human Language

• Incompletely Regular (Both Regular and Idiosyncratic)
• Compositional (or Recursive)
• Metaphorical
• Associated with Knowledge
• Interactive
Natural Language Understanding by Computer Is Extremely Difficult

• It is still not clear whether it is possible to realize human language ability on computer

• On modern computer
  – The incomplete regularity and compositionality characteristics imply complex combinatorial computation
  – The metaphor, knowledge, and interaction characteristics imply exhaustive computation

• Big question: can we invent new computer closer to human brain?
Reason of Challenge

• A computer system must be constructed based on math
• Open question: whether it is possible to process natural language as humans, using math models
• Natural language processing is believed to be AI complete
Simplified Problem Formulation
- Eg., Question Answering

Question answering, including search, can be practically performed, because it is simplified.
Data-driven Approach May Work

• Hybrid is most realistic and effective for natural language processing, and AI
  – machine learning based
  – human-knowledge incorporated
  – human brain inspired

• Big data and deep learning provides new opportunity
Advancement in AI, including NLP can be made through the closed loop.
Fundamental Problems of Statistical Natural Language Processing

• Classification: assigning a label to a string
  \[ S \rightarrow C \]

• Matching: matching two strings
  \[ s, t \rightarrow \mathbb{R}^+ \]

• Translation: transforming one string to another
  \[ S \rightarrow t \]

• Structured prediction: mapping string to structure
  \[ S \rightarrow S' \]

• Markov decision process: deciding next state given previous state and action
  \[ \text{D} \]
Fundamental Problems of Statistical Natural Language Processing

• Classification
  – Text classification
  – Sentiment analysis

• Matching
  – Search
  – Question answering
  – Dialogue (single turn)

• Translation
  – Machine translation
  – Speech recognition
  – Hand writing recognition
  – Dialogue (single turn)

• Structured Prediction
  – Named entity extraction
  – Part of speech tagging
  – Sentence parsing
  – Semantic parsing

• Markov Decision Process
  – Dialogue (multi turn, task dependent)
Lower Bound of User Need vs Upper Bound of Technology

Pushing Upper Bound of Technology
Deep Learning for Natural Language Processing (DL for NLP)

• State-of-Art Performances in
  – Classification
  – Matching
  – Translation
  – Structured Prediction

• Particularly, Neural Machine Translation outperforms Statistical Machine Translation
References

1. 李航，迎接自然语言处理新时代，计算机学会通讯，2017年第2期
2. 李航，简论人工智能，计算机学会通讯，2016年第5期
3. 李航，对于AI我们应该期待什么，计算机学会通讯，2016月第11期
4. 李航，技术的上界与需求的下界，新浪博客，2014年
Outline of Lecture

• Introduction
• Basics of DL for NLP
• State of the Art of DL for NLP
• Previous Work at Noah’s Ark Lab
• Recent Progress at Noah’s Ark Lab
• Advantages and Disadvantages
• Summary
Basics of DL for NLP

- Word Embedding
- Convolutional Neural Network (CNN)
- Recurrent Neural Network (RNN)
- Sequence-to-Sequence Learning
Word Embedding
Word Embedding

- Motivation: representing words with low-dimensional real-valued vectors, utilizing them as input to deep learning methods, vs one-hot vectors
- Method: SGNS (Skip-Gram with Negative Sampling)
- Tool: Word2Vec
- Input: words and their contexts in documents
- Output: embeddings of words
- Assumption: similar words occur in similar contexts
- Interpretation: factorization of mutual information matrix
- Advantage: compact representations (usually 100~ dimensions)
Skip-Gram with Negative Sampling (Mikolov et al., 2013)

- Input: occurrences between words and contexts

<table>
<thead>
<tr>
<th></th>
<th>$C_1$</th>
<th>$C_2$</th>
<th>$C_3$</th>
<th>$C_4$</th>
<th>$C_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_1$</td>
<td>5</td>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>$w_2$</td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>$w_3$</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Probability model:

\[
P(D = 1 \mid w, c) = \sigma(\tilde{w} \cdot \tilde{c}) = \frac{1}{1 + e^{-\tilde{w} \cdot \tilde{c}}}
\]

\[
P(D = 0 \mid w, c) = \sigma(-\tilde{w} \cdot \tilde{c}) = \frac{1}{1 + e^{\tilde{w} \cdot \tilde{c}}}
\]
Skip-Gram with Negative Sampling

• Word vector and context vector: lower dimensional (parameter ) vectors $\vec{w}, \vec{c}$
• Goal: learning of the probability model from data
• Take co-occurrence data as positive examples
• Negative sampling: randomly sample $k$ unobserved pairs $(w, c_N)$ as negative examples
• Objective function in learning

$$L = \sum_{w} \sum_{c} \#(w, c) \log \sigma(\vec{w} \cdot \vec{c}) + k \cdot E_{c_N \sim P} \log \sigma(-\vec{w} \cdot \vec{c}_N)$$
• Algorithm: stochastic gradient descent
Interpretation as Matrix Factorization (Levy & Goldberg 2014)

- Pointwise Mutual Information Matrix

\[ \log \frac{P(w, c)}{P(w)P(c)} \]
Interpretation as Matrix Factorization

\[ M = WC^T \]

Matrix factorization, equivalent to SGNS

Word embedding
Convolutional Neural Network
Convolutional Neural Network

- Motivation: representing sequence of words and utilizing the representation in deep learning methods
- Input: sequence of word embeddings, denoting sequence of words (e.g., sentence)
- Output: representation of input sequence
- Learning of model: stochastic gradient descent
- Advantage: robust extraction of n-gram features; can be used as part of deep model for sequence processing (e.g., sentence classification)
Convolutional Neural Network (CNN) (Kim 2014, Blunsom et al. 2014, Hu et al., 2014)

- Concatenation
  - Shared parameters on same level
  - Fixed length, zero padding

- Max pooling
  - Convolution

```
the    cat    sat    on    the    mat
```

```
the    cat
sat    on
the    cat    sat
```

```
sat    on
the    mat
on    the    mat
```

```
the    cat
cat    sat
the    cat    sat
```

```
cat    sat
sat    on
sat    on
```

```
the    mat
on    the
```

```
the    mat
```

```
mat
```
Example: Image Convolution

Dark Pixel Value = 1, Light Pixel Value = 0
Dot in Filter = 1, Others = 0

Leow Wee Kheng
Example: Image Convolution

Convoluion Operation

- Scanning image with filter having 3*3 cells, among them 3 are dot cells
- Counting number of dark pixels overlapping with dot cells at each position
- Creating feature map (matrix), each element represents similarity between filter pattern and pixel pattern at one position
- Equivalent to extracting feature using the filter
- Translation-invariant
Convolution

\[ z^{(l,f)}_i = \sigma(w^{(l,f)} \cdot z^{(l-1)}_i + b^{(l,f)}) \quad f = 1, 2, \ldots, F_l \]

- \( z^{(l,f)}_i \) is output of neuron of type \( f \) for location \( i \) in layer \( l \)
- \( w^{(l,f)}, b^{(l,f)} \) are parameters of neuron of type \( f \) in layer \( l \)
- \( \sigma \) is sigmoid function

- \( z^{(l-1)}_i \) is input of neuron for location \( i \) from layer \( l - 1 \)
- \( z^{(0)}_i \) is input from concatenated word vectors for location \( i \)

\[ z^{(0)}_i = [x^T_i, x^T_{i+1}, \ldots, x^T_{i+h-1}]^T \]
Max Pooling

\[ z_{i}^{(l,f)} = \max( z_{2i-1}^{(l-1,f)}, z_{2i}^{(l-1,f)} ) \]

- \( z_{i}^{(l,f)} \) is output of pooling of type \( f \) for location \( i \) in layer \( l \)
- \( z_{2i-1}^{(l-1,f)}, z_{2i}^{(l-1,f)} \) are input of pooling of type \( f \) for location \( i \) in layer \( l \)

Equivalent to n-gram feature selection
Sentence Classification Using Convolutional Neural Network

\[ y = f(x) = \text{soft max}(Wz + b) \]

\[ z = \text{CNN}(x) \]
Recurrent Neural Network
Recurrent Neural Network

- Motivation: representing sequence of words and utilizing the representation in deep learning methods
- Input: sequence of word embeddings, denoting sequence of words (e.g., sentence)
- Output: sequence of internal representations (hidden states)
- Variants: LSTM and GRU, to deal with long distance dependency
- Learning of model: stochastic gradient descent
- Advantage: handling arbitrarily long sequence; can be used as part of deep model for sequence processing (e.g., language modeling)
Recurrent Neural Network (RNN) (Mikolov et al. 2010)

\[ h_t = f(h_{t-1}, x_t) \]

\[ h_t \rightarrow h_{t-1} \]

\[ x_t \rightarrow h_t \]

\[ h_{t-1} \rightarrow h_{t-1} \]

\[ h_t \rightarrow \text{the cat sat on the mat} \]

\[ h_t \rightarrow \text{the} \]

\[ h_t \rightarrow \text{cat} \]

\[ h_t \rightarrow \text{sat} \]

\[ h_t \rightarrow \text{mat} \]
Recurrent Neural Network

\[ h_t = f(h_{t-1}, x_t) = \tanh(W_h h_{t-1} + W_x x_t + b_{hx}) \]
Long Term Short Memory (LSTM)  
(Hochreiter & Schmidhuber, 1997)

- A memory (vector) to store values of previous state
- Input gate, output gate, and forget gate to control
- Gate: element-wise product with vector of values in [0,1]

\[
i_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i) \\
f_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f) \\
o_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o) \\
g_t = \tanh(W_{gh}h_{t-1} + W_{gx}x_t + b_g) \\
c_t = i_t \otimes g_t + f_t \otimes c_{t-1} \\
h_t = o_t \otimes \tanh(c_t)
\]
Gated Recurrent Unit (GRU)  
(Cho et al., 2014)

- A memory (vector) to store values of previous state
- Reset gate and update gate to control

\[
\begin{align*}
    r_t &= \sigma(W_{rh} h_{t-1} + W_{rx} x_t + b_r) \\
    z_t &= \sigma(W_{zh} h_{t-1} + W_{zx} x_t + b_z) \\
    g_t &= \tanh(W_{gh}(r_t \otimes h_{t-1}) + W_{gx} x_t + b_g) \\
    h_t &= z_t \otimes h_{t-1} + (1 - z_t) \otimes g_t
\end{align*}
\]
Recurrent Neural Network Language Model

Model

\[ h_t = \tanh(W_h h_{t-1} + W_x x_t + b_{hx}) \]

\[ p_t = P(x_t \mid x_1 \cdots x_{t-1}) = \text{soft max}(Wh_t + b) \]

Objective of Learning

\[ \frac{1}{T} \sum_{t=1}^{T} - \log \hat{p}_t \]

• Input one sequence and output another
• In training, input sequence is same as output sequence
Sequence to Sequence Learning
Translation: Sequence to Sequence Learning

- Hierarchical LSTM
- Different LSTM models for encoder and decoder
- Reverse order of words in source sentence

\[
P(y_t | y_1 \cdots y_{t-1}, x) = g(y_{t-1}, s_t)
\]
\[
h_t = f_e(x_t, h_{t-1}), s_t = f_d(y_{t-1}, s_{t-1})
\]

• Sutskever et al. 2014
Translation: RNN Encoder-Decoder

\[ P(y_t \mid y_1 \cdots y_{t-1}, x) = g(y_{t-1}, s_t, c), c = h_T \]
\[ s_t = f_d(y_{t-1}, s_{t-1}, c) \]
\[ h_t = f_e(x_t, h_{t-1}) \]

- Context vector represents source sentence
- GRU is used
- Cho et al. 2014
Translation: Attention Mechanism

\[ P(y_t | y_1 \cdots y_{t-1}, x) = g(y_{t-1}, s_t, c_t) \]
\[ s_t = f_d(y_{t-1}, s_{t-1}, c_t) \]
\[ h_t = f_e(x_t, h_{t-1}) \]
\[ c_t = \sum_{j=1}^T \alpha_{ij} h_j \]
\[ \alpha_{ij} = q(s_{i-1}, h_j) \]

- Context vector represents attention
- Corresponds to alignment relation
- Encoder: Bidirectional RNN

Bahdanau, et al. 2014
Outline of Lecture

• Introduction
• Basics of DL for NLP
• *State of the Art of DL for NLP*
• Previous Work at Noah’s Ark Lab
• Recent Progress at Noah’s Ark Lab
• Advantages and Disadvantages
• Summary
State of the Art of DL for NLP

• Classification
• Matching
• Translation
• Structured Prediction

• References can be found at Hang Li, Zhengdong Lu, SIGIR 2016 Tutorial
Classification

• Examples of Tasks
  – **Search**: query classification, document classification
  – **Question Answering**: question classification, answer classification

• Approaches
  – World Level Model
  – Character Level Model
  – Hierarchical Model (for document classification)
Sentence Classification: Word Level Model

Classifier:
Softmax

Neural Network:
Convolutional Neural Network, Deep Neural Network

Input:
Continuous Word Embedding, Discrete Word Embedding (one-hot)

- Kim 2014
- Blunsom et al. 2014
- Johnson & Zhang 2015
- Iyyer et al. 2015
Document Classification: Character Level Model

Neural Network 1: Deep Convolutional Neural Network

Neural Network 2: 3-Layer Fully-Connected Neural Network

Input: Character Embedding

Data: Large Scale Training Dataset

Class: Semantic Topics

• Zhang et al. 2016
Document Classification: Hierarchical Model

Classifier:
- Softmax

First Layer Network:
- Recurrent Neural Network (LSTM, GRU)

Second Layer Network:
- Recurrent Neural Network (LSTM, GRU)

Attention:
- Can be Employed between Two Layers

- Tang et al. 2015
- Lai et al. 2015
- Yang et al. 2016
Matching

• Examples of Tasks
  – **Search**: query-document (title) matching, similar query finding
  – **Question Answering**: question answer matching

• Approaches
  – Projection to Latent Space
  – One Dimensional Matching
  – Two Dimensional Matching
  – Tree Matching
Matching: Projection to Latent Space

• Natural extension of Vector Space Model

Latent Space

Dot Product (Score)

Question Representation

Answer Representation

Neural Networks:
  Convolutional Neural Network
  Deep Neural Network
  Recurrent Neural Network

• Huang et al. 2013
• Shen et al. 2014
• Severyn & Moschitti 2015
Matching: One Dimensional Matching

Neural Network 1:
Convolutional Neural Network

Neural Network 2:
Deep Neural Network, Tensor Network

- Hu et al. 2014
- Qiu & Huang 2015
Matching: Two Dimensional Matching

Neural Network 1:
Convolutional Neural Network

Neural Network 2:
Deep Neural Network

• Hu et al. 2014
• Pang et al. 2016
• Wan et al. 2016
Matching: Tree Matching

DNN with Sparse Input Layer

Tree Matching Representation

Matching Pattern Identification

E.g. Q="how is X[city]", A="The food in X[city] is great"

Question Parse Tree

Answer Parse Tree

Parser

Question

Score

Answer

Parser

• Wang et al. 2015
Translation

• Examples of Tasks
  – **Machine Translation:** sentence level translation
  – **Question Answering:** answer generation from question
  – **Search:** similar query generation

• Approaches
  – Sequence-to-Sequence Learning
  – RNN Encoder-Decoder
  – Attention Mechanism
Translation: Sequence-to-Sequence Learning (Same for RNN Encoder-Decoder)

Encoder:
Recurrent Neural Network

Decoder:
Recurrent Neural Network

- Sutskever et al. 2014
- Cho et al. 2014
Translation: Sequence-to-Sequence Learning

Encoder:
Recurrent Neural Network

Decoder:
Recurrent Neural Network

Attention Mechanism

Bahdanau, et al. 2014
Structured Prediction

• Examples of Tasks
  – **Search**: named entity recognition in query and document
  – **Question Answering**: named entity recognition in question and answer

• Approaches
  – CNN
  – Sequence-to-Sequence Learning
  – Neural Network based Parsing

- Diagram:
  - Deep Neural Network
  - Named Entities, etc
  - Sentence
Structured Prediction: CNN

Classifier at Each Position:
Softmax
Neural Network:
Convolutional Neural Network

Label at Each Position

Sentence Representation

Neural Network

Sentence

• Collobert et al. 2011
Structured Prediction: Sequence-to-Sequence Learning

**Neural Network:**
Sequence-to-Sequence Learning Model

**Training Data:**
Pairs of Sentence and Linearized Parse Tree

E.g.,
John has a dog . \( \rightarrow (S (NP NNP)_{NP} (VP VBZ (NP DT NN)_{NP})_{VP} .)_{S} \)

- Vinyals et al. 2015
Structured Prediction: Neural Network based Parsing

Parser:
- Transition-based Dependency Parser
- Constituency Parser
- CRF Parser

Neural Network:
- Deep Neural Networks

Training Data:
- Pairs of Sentence and Parse Tree

• Chen & Manning, 2014
• Durrett & Klein, 2015
• Zhou et al., 2015
• Andor et al., 2016
Outline of Lecture

• Introduction
• Basics
• State of the Art
• Previous Work at Noah’s Ark Lab
• Recent Progress at Noah’s Ark Lab
• Advantages and Disadvantages
• Summary
Question Answering
- DeepMatch CNN
Demo
Retrieval based Question Answering System

- **Retrieval**: Retrieved Questions and Answers
- **Matching**: Matched Answers
- **Ranking**: Ranked Answers

**Online**
- Question
- Retrieval
- Matching Models
- Ranking Model
- Best Answer

**Offline**
- Index of Questions and Answers
Deep Match Model CNN

- Represent and match two sentences simultaneously
- Two dimensional model
- State of art model for matching in question answering
Image Retrieval
- Multimodal CNN
Demo
Multimodal CNN

- One Convolutional Neural Network represents image
- One Convolutional Neural Network represents text
- Multi Layer Perceptron conducts matching

![Diagram of multimodal CNN with a dog catching a ball and a sentence structure diagram with nodes labeled as 'a', 'dog', 'is', 'catching', 'a', 'ball']
Experimental Results

• Experiment
  – Trained with 30K Flickr data
  – Outperforming other state-of-the-art models
Natural Language Dialogue
- Neural Responding Machine
Demo
Neural Responding Machine

- Using both local and global attention mechanisms
Experimental Results

• Experiment
  – Trained with 4.4 million Weibo data (Chinese)
  – 95% of responses are natural, 76% of responses are appropriate as replies

<table>
<thead>
<tr>
<th>Message</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Occupy Central is finally over</td>
<td>Will Lujiazui (finance district in Shanghai) be the next?</td>
</tr>
<tr>
<td>I want to buy a Samsung phone</td>
<td>Let us support our national brand.</td>
</tr>
</tbody>
</table>
Neural Machine Translation
Neural Machine Translation

A cat is sitting on the mat

• Using coverage vectors to avoid over-translation and under-translation
• Using context gates to dynamically control the impact of attention

Encoder

Decoder

一只猫坐在那张垫子上
• Using coverage vectors to avoid over-translation and under-translation

A cat is sitting on the mat

Encoder

Decoder

一只猫坐在那张垫子上
Experimental Results

• Experiment
  – Trained with 1.25 million LDC data (Chinese-English)

C-E Translation

- 有一些恐怖袭击会愈演愈烈。
- NMT There will be some terrorist attacks.
- +Both Some terrorist attacks will become more and more intense.
Experimental Result

- Google NMT system works better, apparently due to its larger training data and more powerful computing architecture.
- Google NMT system also employs coverage mechanism.
Outline of Lecture

• Introduction
• Basics of DL for NLP
• State of the Art of DL for NLP
• Previous Work at Noah’s Ark Lab
• Recent Progress at Noah’s Ark Lab
• Advantages and Disadvantages
• Summary
Question Answering
- Neural Enquirer
**Question Answering from Relational Database**

**Q:** How many people participated in the game in Beijing?

**A:** 4,200

**SQL:** `select #_participants, where city=beijing`

**Q:** When was the latest game hosted?

**A:** 2012

**SQL:** `argmax(city, year)`

**Q:** Which city hosted the longest Olympic game before the game in Beijing?

**A:** Athens

<table>
<thead>
<tr>
<th>year</th>
<th>city</th>
<th>#_days</th>
<th>#_medals</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>Sydney</td>
<td>20</td>
<td>2,000</td>
</tr>
<tr>
<td>2004</td>
<td>Athens</td>
<td>35</td>
<td>1,500</td>
</tr>
<tr>
<td>2008</td>
<td>Beijing</td>
<td>30</td>
<td>2,500</td>
</tr>
<tr>
<td>2012</td>
<td>London</td>
<td>40</td>
<td>2,300</td>
</tr>
</tbody>
</table>
Neural Enquirer

- Query Encoder: encoding query
- Table Encoder: encoding entries in table
- Five Executors: executing query against table

Conducting matching between question and database entries multiple times
• Creating query embedding using RNN
• Creating table embedding for each entry using DNN
• Five layers, except last layer, each layer has reader, annotator, and memory
• Reader fetches important representation for each row, e.g., city=beijing
• Annotator encodes result representation for each row, e.g., row where city=beijing
Experimental Results

• Experiment
  – Olympic database
  – Trained with 25K and 100K synthetic data
  – Accuracy: 84% on 25K data, 91% on 100K data
  – Significantly better than SemPre (semantic parser)
  – Criticism: data is synthetic

<table>
<thead>
<tr>
<th></th>
<th>25K Data</th>
<th>100K Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Semantic Parser</td>
<td>End-to-End</td>
</tr>
<tr>
<td></td>
<td>Semantic Parser</td>
<td>End-to-End</td>
</tr>
<tr>
<td>25K Data</td>
<td>65.2%</td>
<td>84.0%</td>
</tr>
<tr>
<td>100K Data</td>
<td>NA</td>
<td>90.6%</td>
</tr>
</tbody>
</table>
Question Answering
- GenQA
Q: How tall is Yao Ming?
A: He is 2.29m tall and is visible from space.
\( (Yao\ Ming, \ height, \ 2.29m) \)

Q: Which country was Beethoven from?
A: He was born in what is now Germany.
\( (Ludwig\ van\ Beethoven, \ place\ of\ birth, \ Germany) \)

Q: How tall is Liu Xiang?
A: He is 1.89m tall
GenQA

- **Interpreter:** creates representation of question using RNN
- **Enquirer:** retrieves top k triples with highest matching scores using CNN model
- **Generator:** generates answer based on question and retrieved triples using attention-based RNN
- **Attention model:** controls generation of answer

**Key idea:**
- Generation of answer based on question and retrieved result
- Combination of neural processing and symbolic processing
Retaining both symbolic representations and vector representations
• Using question words to retrieve top $k$ triples
• Calculating matching scores between question and triples using CNN model
• Finding best matched triples

Enquirer: Retrieval and Matching

Retrieved Top $k$ Triples and Embeddings

< liu xiang, height, 1.90m>
< yao ming, height, 2.26m>
... ...
<liu xiang, birth place, shanghai>
• Generating answer using attention mechanism
• At each position, a variable decides whether to generate a word or use the object of top triple
Experimental Results

• Experiment
  – Trained with 720K question-answer pairs (Chinese) associated with 1.1M triples in knowledge-base, *data is noisy*
  – Accuracy = 52%
  – Data is still noisy

<table>
<thead>
<tr>
<th>Question</th>
<th>Answer</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Who wrote the Romance of the Three Kingdoms?</td>
<td>Luo Guanzhong in Ming dynasty</td>
<td><strong>correct</strong></td>
</tr>
<tr>
<td>How old is Stefanie Sun this year?</td>
<td>Thirty-two, he was born on July 23, 1978</td>
<td><strong>wrong</strong></td>
</tr>
<tr>
<td>When will Shrek Forever After be released?</td>
<td>Release date: Dreamworks Pictures</td>
<td><strong>wrong</strong></td>
</tr>
</tbody>
</table>
Natural Language Dialogue
- CopyNet
Single Turn Dialogue with Generating and Copying Mechanism

Dialogue system can not only generate response, but also copy from given message.
CopyNet can either generate word based on attentive read, or copy word based on selective read.
Characteristics: CopyNet

- Decoder can both generate and copy
- Mixture model of generating and copying
- Attentive read: find suitable word to influence generation of word in target sequence
- Selective read: find location of word to be copied from source sequence
- Model is fully differentiable
- Training: maximum likelihood of target sequence given source sequence
Experimental Results

• Experiment
  – Summarization of short text in Chinese
  – Trained with 2.4M text-summary pairs
  – Tested with 9.3K text-summary pairs

<table>
<thead>
<tr>
<th>Model</th>
<th>ROUGE-1</th>
<th>ROUGE-2</th>
<th>ROUGE-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>RNN -C</td>
<td>29.9</td>
<td>17.4</td>
<td>27.2</td>
</tr>
<tr>
<td>RNN -W</td>
<td>26.8</td>
<td>16.1</td>
<td>24.1</td>
</tr>
<tr>
<td>CopyNet -C</td>
<td>34.4</td>
<td>21.6</td>
<td>31.3</td>
</tr>
<tr>
<td>CopyNet -W</td>
<td>35.0</td>
<td>22.3</td>
<td>32.0</td>
</tr>
</tbody>
</table>
Outline of Lecture

• Introduction
• Basics
• State of the Art
• Previous Work at Noah’s Ark Lab
• Recent Progress at Noah’s Ark Lab
• Advantages and Disadvantages
• Summary
Advantages and Disadvantages of DL

- **Strength**
  - Good at *pattern recognition* problems
  - Data-driven, performance is high in many tasks
  - End-to-end training, little or no domain knowledge is needed in system construction
  - Representation learning, possible in cross modal processing
  - Gradient-based learning, learning algorithm is simple
  - Powerful for supervised learning setting

- **Weakness**
  - Not good at *inference and decision* problems
  - Data-hungry and thus is not suitable when data size is small
  - Difficult to handle tail phenomena
  - Model is usually a black box and is difficult to understand
  - Computational cost of learning is high
  - Unsupervised learning methods are needed
  - Still lack of theoretical foundation
End-to-End Learning
Generation-based Dialogue

Encoder

Internal Representation

Decoder

Target Sentence

Source Sentence

Attention Mechanism
Representation Learning
Symbolic Matching Models

Vector Space Model, BM25, Language Model for IR

Linear Projection into Latent Space
Wu et al. 2012
Neural Matching Models

Neural matching models are natural extension of symbolic matching models

Multimodal Match Model (CNN), Ma et al. 2015
Challenge in Tail
Natural Language Processing Problems are Non-Parametric

How to deal with the long tail is a challenging issue.

Xinhua News Data

Vocabulary size increases when data size increases

Data from Xiao Chen
Theoretical Analysis
Generalization Ability of Deep Learning

• In practice, usually both training errors and test errors are small, i.e., no over-fitting
• Neural networks can “memorize” training instances
• On the other hand, neural networks can over-fit (i.e., test errors are large although training errors are small), if random noise is injected into training data
• Number of parameters is larger than number of training instances
• Many open questions about the learning ability of deep learning
Inference and Decision
## Comparison between Single-turn QA and Multi-turn QA by Humans

<table>
<thead>
<tr>
<th>Single-turn QA</th>
<th>Multi-turn QA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Q:</strong> How tall is Yao Ming?</td>
<td><strong>Q:</strong> How tall is Yao Ming?</td>
</tr>
<tr>
<td><strong>A:</strong> He is 2.29m tall.</td>
<td><strong>A:</strong> He is 2.29m tall.</td>
</tr>
<tr>
<td><strong>Q:</strong> Who is taller, Yao Ming or Liu Xiang?</td>
<td><strong>Q:</strong> Who is taller, Yao Ming or Liu Xiang?</td>
</tr>
<tr>
<td><strong>A:</strong> He is taller, and I think that Liu Xiang is only 1.89m tall.</td>
<td><strong>A:</strong> He is taller, and I think that Liu Xiang is only 1.89m tall.</td>
</tr>
</tbody>
</table>

• Single turn QA is only related to fact retrieval and answer generation.
• Multi-turn QA needs fact retrieval and answer generation, as well as other *mental processing*. More modules in human brain are involved.
Deep Learning and Multi-turn Dialogue

• DL may not be enough for natural language dialogue
• Key is dialogue management, including dialogue control and dialogue modeling
• Involvement of multiple “modules”, each having multiple “states”
• Recent work tries to use reinforcement learning
• There are many open questions
Outline of Lecture

• Introduction
• Basics of DL for NLP
• State of the Art of DL for NLP
• Previous Work at Noah’s Ark Lab
• Recent Progress at Noah’s Ark Lab
• Advantages and Disadvantages
• Summary
Summary

• Deep Learning brings high performance in fundamental language processing problems, particularly translation

• Basic models: Word Embedding, CNN, RNN, Sequence to Sequence Learning

• Deep neural network models are state of the art for question answering, image retrieval, generation based dialogue, machine translation

• Recent progress, combination of symbolic and neural processing

• Advantages and limitations
References


References


Thank you!