Deep Learning for Question Answering

Lei LI

Toutiao Lab

12/3/16
Goal

Enable machines to comprehend and converse
• Bots to write/tell a story
• Bots to chitchat
• Bots to organize knowledge
Major applications of QA

- Search engine
- Personal assistant
- Information platform
Information consumption platform

News article
Stories
Video
Community QA
Three key areas in effective information consumption

Creating intelligent machines that understand information in depth (text, images, videos, comments, etc.) to better serve our users with what they like.

Developing large scale machine learning algorithms for personalized information recommendation.
QA can be one genre of content
Outline

- Problem Setup, Knowledge graphs
- Basic DL techniques
- Word, entity and relation embeddings
- Recurrent neural nets for processing sequence
- Focused Pruning: parsing the subject mention
- Finding the right relation and subject entity
- Other approaches:
  - LTG+CNN
  - Memory network
Categories of Questions

• Factoid: who is the president of USA?
• Descriptive: what are characteristics of the new Mac Pro
• Procedural: how to install windows 10
• Calculation: how many Chinese won Turing awards?
• Causal: why is it dark at night
• Opinion: how do you think about Trump?
Factoid questions: Simple to Complex

Simple Question
- One that can be answered with single evidence
- E.g. Who wrote the book of Beijing Folding?

Multi-hop Question
- Requires with many facts
- E.g.

Aggregate Question
- Requires with many facts and calculation
- E.g. what is the longest Olympic opening before Beijing 2008

This tutorial
Q: Where was David Beckham born?
What do we need to answer questions?

• Fact storage: knowledge graph
• Mapping from natural questions to structured queries executable on knowledge graph

Q: Where was David Beckham born?
Knowledge Graph

Knowledge as triples

<DavidBeckham, Name, "David Beckham">
<DavidBeckham, PlaceOfBirth, Leytonstone>

<Subject, relation, object>
Structure Query on KG

SPARQL

SELECT ?object
WHERE { <DavidBeckham> <PlaceOfBirth> ?object }
From natural language question to structured query

Question

Where was David Beckham born?

SPARQL query

SELECT ?object
WHERE { <DavidBeckham> <PlaceOfBirth> ?object }

related work

[Berant 2013]
[Yih 2014]
[Bordes 2015]
Why difficult for machines?

Language complexity
- 奥巴马总统在哪儿生的？
- 奥巴马总统出生地在哪里？
- What is the birthplace of Mr. Obama?
- Where was Mr. Obama born?

Ambiguity
- 麦克乔丹是谁？
- Who is Michael Jordan?

Sparse Label
- 2千万事实，十万标注问答对
- 22 million, 100k labeled QA pairs
Simple solutions: N-gram

• Rank and match all possible n-grams in the question
• Link them to entities in KG via alias matching

Where was David Beckham born?

N-gram candidates:
Uni-gram: Where, was, David, Beckham, born, ?
Bi-gram: Where was, was David, David Beckham, Beckham born, born ?
Tri-gram: Where was David, was David Beckham, David Beckham born, Beckham born ?
Four-gram: Where was David Beckham, was David Beckham born, David Beckham born ?
Improved simple solutions

• Rank and match all possible n-grams in the question
• Prune the n-grams with heuristics
• Link them to entities in KG via alias matching

Where was David Beckham born?

N-gram candidates:
Uni-gram: Where, was, David, Beckham, born, ?
Bi-gram: Where was, was David, David Beckham, Beckham born, born ?
Tri-gram: Where was David, was David Beckham, David Beckham born, Beckham born ?
Four-gram: Where was David Beckham, was David Beckham born, David Beckham born ?
Challenges

1. Insufficient Knowledge Representation

• Where is San Francisco?
• What is Columbus famous for?

• MORE than 400 entities
• City, County, Person, Movie, etc

2. Too Much Noise from N-Grams

• What theme is the book the armies of memory?

• the book: 73
• theme: 252
• memory: 553
• ……

Confidential
Outline

- Problem Setup, Knowledge graphs
- Basic DL techniques
- Word, entity and relation embeddings
- Recurrent neural nets for processing sequence
- Focused Pruning: parsing the subject mention
- Finding the right relation and subject entity
- Other approaches:
  - LTG+CNN
  - Memory network
DL algorithms work well for

Supervised learning

data

\[ X \xrightarrow{f(\cdot)} Y \]

label

Cat/dog/...

“今天天气不错！”

“A giraffe standing next to forest

“Today is a nice day”

“打车去故宫”
Inspired by a biological neuron

Neural networks: massively connected simple units

Image credit:
How to model a single artificial neuron?

\[ \sum \text{Transfer function} \]

- \( x_1 \) with weight \( w_1 \)
- \( x_2 \) with weight \( w_2 \)
- \( x_3 \) with weight \( w_3 \)

Activation function
Activation functions

\[
\tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1} \quad \text{relu}(x) = \max(0, x)
\]

Useful for modeling probability (in classification task)
Supervised Learning with Neural Nets

Input

\[ h_1 = \text{tanh}(w_1 \cdot x + b_1) \]

\[ y = \text{softmax}(w_2 \cdot h_1 + b_2) \]
Numerical Example

\[ h_1 = \tanh(w_1 \cdot x + b_1) \]
\[ y = \text{softmax}(w_2 \cdot h_1 + b_2) \]

Input

\[ \begin{bmatrix}
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & \ldots
\end{bmatrix} \]

\[ w_1 : 256 \times 784 \]
Objective / Loss: cross-entropy

The objective function for cross-entropy loss is given by:

\[ l(f(x_i), y_i) = -\log f(x_i)_{y_i} \]

where \( f(x_i) \) is a vector (e.g. \( \in \mathbb{R}^{10} \)), representing the predicted distribution. \( y_i \) is the ground-truth label, which can be represented as an one-hot “distribution” \([0,...,0,1,0,...,0]\).

The cross-entropy function is defined as:

\[ H(p, q) = -\sum_k p_k \log q_k \]
Cross-entropy

\[ H(p, q) = - \sum_k p_k \log q_k \]

Average number of bits needed to represent message in q, while the actual message is distributed in p

OR. roughly

The information gap between p and q + (some const)

Minimizing cross-entropy == diminishing the information gap

\[ H(y_i, f(x_i)) = - \sum_k y_{i,k} \log f(x_i)_k = - \log f(x_i)_{y_i} \]

Ideal case \( f(x_i)_{y_i} \) => 1.0
Alternative View: Max cond. log-likelihood

\[
\begin{align*}
\max \log p(y_i|x_i; w) &= \sum_k y_{i,k} \log f(x_i)_k \\
\text{Or equivalently} & \\
\min - \sum_k y_{i,k} \log f(x_i)_k
\end{align*}
\]
Deep Neural Nets

\[ h_1 = \sigma_1(w_1 \cdot x + b_1) \quad h_2 = \sigma_2(w_2 \cdot h_1 + b_2) \]
Training DNN

Given: N data points 
\((x_1, y_1) \ldots (x_N, y_N)\)

Goal: find the best model parameter w, to minimize cost

\[
L(w) = \sum_{i=1}^{N} l(f(x_i, w), y_i)
\]
Training deep neural nets

Stochastic gradient descent algorithm

for iteration 1 to N (or until convergence)
compute $g = \frac{\partial}{\partial w_j}$

$w = w - a \cdot g$

To improve efficiency:
Mini-Batch
Compute gradient and update parameters for every batch of $k$ data samples.

Advanced alg: Momentum, Adagrad, Adam, ...

Step size
gradient
Forward and Backward propagation

forward pass: computing network prediction
\[ h_i = \sigma_i (w_i \cdot h_{i-1}) \]

backward prop: computing gradient from layer-wise error
\[ \delta_{i-1} = w_i^T \cdot (\delta_i \odot \sigma'_i) \]
\[ \frac{\partial}{\partial w_j} = h_{i-1} \cdot \delta_i^T \]
More variation

• Optimization algorithms
  – Momentum
  – Adagrad
  – Adadelta
  – Adam

• Dropout
  – Randomly zeros the output neurons in each layer

• Regularization
  – L1, L2, to improve generalization
Deep Learning platform

- Tensorflow (Google)
- Torch (NEC, FB)
- Caffe (ucb)
- Theano (U. Montreal)
- MXNet (DMLC, Li Mu et al)

- Provides easy language to construct network
- Rich set of layers, with forward and backward steps
- Library of optimization algorithms
- Many research papers build models based on these
Outline

• Problem Setup, Knowledge graphs
• Basic DL techniques
• Word, entity and relation embeddings
• Recurrent neural nets for processing sequence
• Focused Pruning: parsing the subject mention
• Finding the right relation and subject entity
• Other approaches:
  – LTG+CNN
  – Memory network
How to represent characters and words

Where was David Beckham born?

Well-known methods: word2vec, Glove, etc.
Basic DL technology for language understanding

- Neural Language Model
  - Single layer NN for bigram, [Wei Xu and Alex Rudnicky, 2000]
  - Concatenated Word Embedding to predict next word [Yoshua Bengio, Réjean Ducharme, Pascal Vincent, Christian Jauvin, 2003]
  - RNN Language Model, [Mikolov et al, 2011]

- Basic NLP technology
  - NLP from scratch [Ronan Collebert, Jason Weston et al 2011]
    - WSJ POS 97.29% acc; CoNLL NER 89.59% F1; CoNLL Chunking 94.32% F1
How to represent entities?

5 million entities in cleaned freebase

<DavidBeckham>

1. Random embedding
2. TransE trained embedding
3. zero-training embedding?
Learning entity embedding w/ TransE

<DavidBeckham, PlaceOfBirth, Leytonstone>
Zero-training embedding: Type-vector

- Benefits: no need to train, robust

David Beckham  | 0 0 1 0 1 0 1

Person  | 0 0 1 0 1 0 0
Soccer player  |
Outline

• Problem Setup, Knowledge graphs
• Basic DL techniques
• Word, entity and relation embeddings
• Recurrent neural nets for processing sequence
• Focused Pruning: parsing the subject mention
• Finding the right relation and subject entity
• Other approaches:
  – LTG+CNN
  – Memory network
Challenge in processing language

• How to handle variable length of text sequences?

• Solution:
  – Adding Memory to Computation
Recurrent Neural Networks

Basic version: 1 fixed vector memory

- Remember previous state

Where was David Beckham

\[ h_t = f(W \cdot h_{t-1} + U \cdot x_t) \]

\( f = \text{sigmoid, tanh, relu} \)
Recurrent Neural Networks

- Remember previous state

\[ h_t = f(W \cdot h_{t-1} + U \cdot x_t) \]

\( f = \text{sigmoid}, \text{tanh}, \text{relu} \)
Gated recurrent unit

Adaptively memorize short and long term information

Input: $x_t$  
Memory: $h_t$

\[
\begin{align*}
    r_{t+1} &= \sigma(M_{rx}x_{t+1} + M_{rh}h_t + b_r) \\
    z_{t+1} &= \sigma(M_{zx}x_{t+1} + M_{zh}h_t + b_z) \\
    \tilde{h}_{t+1} &= \tanh(M_{hx}x_{t+1} + M_{hh}(r_{t+1} \otimes h_t) + b_h) \\
    h_{t+1} &= z_{t+1} \otimes \tilde{h}_{t+1} + (1 - z_{t+1}) \otimes h_t
\end{align*}
\]

[Chung et al 2014]
Long-Short Term Memory (LSTM)

Adaptively memorize short and long term information

\[
i_{t+1} = \sigma(M_{ix}x_{t+1} + M_{ih}h_t + b_i)
\]

\[
f_{t+1} = \sigma(M_{fx}x_{t+1} + M_{fh}h_t + b_f)
\]

\[
o_{t+1} = \sigma(M_{ox}x_{t+1} + M_{oh}h_t + b_o)
\]

\[
a_{t+1} = \tanh(M_{cx}x_{t+1} + M_{ch}h_t + b_a)
\]

\[
c_{t+1} = f_{t+1} \otimes c_t + i_{t+1} \otimes a_{t+1}
\]

\[
h_{t+1} = o_{t+1} \otimes \tanh(c_{t+1})
\]
Outline

• Problem Setup, Knowledge graphs
• Basic DL techniques
• Word, entity and relation embeddings
• Recurrent neural nets for processing sequence
• Focused Pruning: parsing the subject mention
• Finding the right relation and subject entity
• Other approaches:
  – LTG+CNN
  – Memory network
Where was David Beckham born?

SELECT ?object
WHERE { <DavidBeckham> <PlaceOfBirth> ?object }
“What theme is the book the armies of memory?”

- the book: 73
- theme: 252
- memory: 553
- ......
Finding subject mention with focused pruning

\[ p(s, r \mid q) = \sum_{f \in \mathcal{F}(q)} p(s, r, f \mid q) \]

Using RNN sequence labelling model

Focus: Where was David Beckham born?

Prob.: 0.05 0.85 0.01
Finding focus ~ sequence labelling

Wuhan Tech University’s nearby handmade noodle house

中心关键词

中心 Wuhan Tech University’s nearby handmade noodle house

如何从上海到杭州开车怎么走

origin destination

上海到杭州开车怎么走

中心关键词

中心 Wuhan Tech University’s nearby handmade noodle house

如何从上海到杭州开车怎么走

origin destination
A Sequence Labelling Task
Named entity recognition

date
In April 1775 fighting broke out between Massachusetts militia units and British regulars at Lexington and Concord. Geo-Political
 Named entity recognition

三藩市长李孟贤 ...  1640 897 45 1890 78 943 3521

B-GPE I-GPE O O B-PER I-PER I-PER

Entity chunking scheme: B-I-O  Begin of entity chunk, In-middle-of entity chunk, Other (not entity)
Traditional approach

• Conditional random fields with rich expert created features.

Features: neighboring words, POS of current word and neighboring words, Lexical features etc.
End-to-end training with minimal linguistic features
Complete NER Model

Chinese NER
OntoNotes Data 4-class:

<table>
<thead>
<tr>
<th>Model</th>
<th>P</th>
<th>R</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bi-NER-WA* Wang et al.</td>
<td>84.42</td>
<td>76.34</td>
<td>80.18</td>
</tr>
<tr>
<td>RNN-2b with WS ours</td>
<td>84.75</td>
<td>77.85</td>
<td>81.15</td>
</tr>
</tbody>
</table>

* Wang et al used bilingual data

OntoNotes Data 18-class:

<table>
<thead>
<tr>
<th>Model</th>
<th>P</th>
<th>R</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sameer Pradhan et al.</td>
<td>78.20</td>
<td>66.45</td>
<td>71.85</td>
</tr>
<tr>
<td>RNN-2b with WS ours</td>
<td>78.69</td>
<td>70.54</td>
<td>74.39</td>
</tr>
</tbody>
</table>

[Zefu Lu, Lei Li, Wei Xu, 2015]
Stacked bi-directional GRU for sentence embedding

Who created the character Harry Potter?

Harry Potter 0.5
Harry 0.2
Potter 0.1
Character 0.05

CreatorOf

CFO: Conditional Focused Neural Question Answering with Large-scale Knowledge Bases
[Zihang Dai, Lei Li, Wei Xu, ACL 2016]
Answers by our CFO system:

**Which school did Harry Potter attend?**
Hogwarts School of Witchcraft and Wizardry
Gregory Primary school

**Who created Harry Potter?**
J.K. Rowling

**Who influenced J.K. Rowling?**
George Eliot
Stephen King

**What books did Stephen King write?**
Las cuatro estaciones/different seasons

[18x441 to 590x536]

[18x441 to 590x536]

[18x441 to 590x536]

[18x441 to 590x536]

[Dai, Li, Xu, 2016]
Does focus help?

- **N-Gram**
  - % Multi-subject cases: 18 / 21 = 85.7%
  - % Single-subject cases: 12051 / 20533 = 58.7%

- **N-Gram+**
  - % Multi-subject cases: 126 / 138 = 91.3%
  - % Single-subject cases: 13460 / 20017 = 67.2%

- **Focused Pruning**
  - % Multi-subject cases: 9925 / 10705 = 92.7%
  - % Single-subject cases: 6482 / 9876 = 65.6%
## Evaluation Results

<table>
<thead>
<tr>
<th>Pruning Method</th>
<th>Relation Network</th>
<th>Entity Representation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Random</td>
</tr>
<tr>
<td>Memory Network [3]</td>
<td>62.9</td>
<td>63.9*</td>
</tr>
<tr>
<td>N-Gram</td>
<td>Embed-AVG</td>
<td>39.4</td>
</tr>
<tr>
<td></td>
<td>LTG-CNN</td>
<td>32.8</td>
</tr>
<tr>
<td></td>
<td>BiGRU</td>
<td>43.7</td>
</tr>
<tr>
<td>N-Gram+</td>
<td>Embed-AVG</td>
<td>53.8</td>
</tr>
<tr>
<td></td>
<td>LTG-CNN[1,2]</td>
<td>46.3</td>
</tr>
<tr>
<td></td>
<td>BiGRU</td>
<td>58.3</td>
</tr>
<tr>
<td>Focused Pruning</td>
<td>Embed-AVG</td>
<td>71.4</td>
</tr>
<tr>
<td></td>
<td>LTG-CNN</td>
<td>67.6</td>
</tr>
<tr>
<td></td>
<td>LTG-CNN+</td>
<td>70.2</td>
</tr>
<tr>
<td></td>
<td>BiGRU</td>
<td>75.2</td>
</tr>
</tbody>
</table>
Comparison

Accuracy

- LTG+CNN: 56.0%
- Memory Network: 62.9%
- CFO: 75.7%

[Reference: Yih et al, ACL 14, Yih et al, ACL 15, Bordes et al 2015]
Conclusion

- Problem Setup, Knowledge graphs
- Basic DL techniques
- Word, entity and relation embeddings
- Recurrent neural nets for processing sequence
- Focused Pruning: parsing the subject mention
- Finding the right relation and subject entity
- Other approaches:
  - LTG+CNN
  - Memory network
Thanks!

Contact: Lei Li (lileilab@toutiao.com)

Joint work with

Zihang Dai (CMU): QA
Wei Xu (Baidu IDL)
Toutiao Lab is Hiring!

Research Scientist and Software Engineer in Machine Learning
Natural Language Understanding
Computer Vision

http://www.toutiao.com/lab
lab-hr@toutiao.com
Reference

Parsing & Sequence labelling
• Collobert et al, Natural language processing almost from scratch
• Lu et al, Twisted recurrent network for named entity recognition
• Huang et al, Bidirectional LSTM-CRF models for sequence tagging
Question Answering


• Antoine Bordes, Nicolas Usunier, Sumit Chopra, Jason Weston, Large-scale Simple Question Answering with Memory Networks, 2015

• Zihang Dai, Lei Li, Wei Xu, CFO: Conditional Focused Question Answering with Large Knowledge-bases. ACL 2016