Short Text Conversation Using Big Data and Deep Learning

Hang Li
Noah’s Ark Lab
Huawei Technologies
Short Text Conversation

Human inputs a message
Computer returns a response

One small step toward the goal of passing Turing test

Alan Turing
Demo: Short Text Conversation

上海今天好热，堪比新加坡。
上海今天热的不一般。

想去武当山 有想同游的么？
我想跟帅哥同游~哈哈

在家门口发现了一片野韭菜！
我看到一片一片的韭菜地.....

芝大综合排名升的好快呀，都第五了
芝大威武
Accuracy Improvement at Noah’s Ark Lab

- 2013/10: 57%
- 2014/3: 64%
- 2014/7: 72%
Outline

• Project: Intelligent Information Assistant
• Short Text Conversation
• Retrieval-based Short Text Conversation
• Learning to Match
• Deep Matching Models
• Summary
Project: Intelligent Information Assistant
Intelligent Information Assistant

- External Information
- Information Recommendation
- Information Management
- Personal Information
- Communications
- Information Extraction
- Question Answering
- Knowledge Base
- Natural Language Conversation
- Human Computer Interaction
Intelligent Help in Huawei Phones
Weibo Robot: 小诺_noah

• Xiaonuo (Weibo robot version 1.0) released on Jan 1, 2013
• Persona: PhD student in NLP
• Number of followees = 523
• Number of followers = 1096
• Features Developed
  – Following people
  – Re-tweeting (forwarding tweets)
  – Generating short comments
Retrieval-based Short Text Conversation
Massive Amount of Data Available

Our paper entitled learning to rank has been accepted by ACL.

We are lucky. Our paper has been accepted by SIGIR this year. We are going to present it.

The PC of WSDM noticed us that our paper has been accepted.

Congratulations! It is a great achievement

Great news! Please accept my congrats!

Awesome! It is a great achievement
System of Short Text Conversation

- Given message, find most suitable response
- Large repository of message-response pairs
- Take it as search problem

Diagram:
- Retrieval -> Retrieved messages and responses
- Matching -> Matched responses
- Ranking -> Ranked responses
- Index of messages and responses
- Learning to match
- Learning to rank
- Best response

Online and offline processes
Learning to Match for Short Text Conversation

Learning System

Matching System

Model

\[ t_1 \quad c_1 \]
\[ t_2 \quad c_2 \]
\[ \vdots \]
\[ t_n \quad c_n \]
\[ t(n+1) \]
Learning to Match
Matching between Heterogeneous Data is Everywhere

• Matching between user and product (collaborative filtering)
• Matching between text and image (image annotation)
• Matching between languages (machine translation)
• Matching between receptor and ligand (drug design)
• Matching between people (dating)
Formulation of Learning Problem

• Learning matching function

\[ f(x, y) \]

• Training data \((x_1, y_1, r_1), \ldots, (x_N, y_N, r_N)\)

• Generated according to

\[ x \sim P(X), \quad y \sim P(Y \mid X), \quad r \sim P(R \mid X, Y) \]
Formulation of Learning Problem

• Loss Function

\[ L(r, f(x, y)) \]

• Risk Function

\[ R(r, f(x, y)) = \int_{X \times Y \times R} P(x, y, r) L(r, f(x, y)) dP(x, y, r) \]

• Objective Function in Learning

\[ \min_{f \in F} \sum_{i=1}^{N} L(r_i, f(x_i, y_i)) + \Omega(f) \]
Matching Problem: Instance Matching

Instances

Can be represented as matching between nodes in bipartite graph
Matching Problem: Feature Matching

Features can be represented as matching between objects in two spaces.

<table>
<thead>
<tr>
<th></th>
<th>y1</th>
<th>y2</th>
<th>y3</th>
<th>yn</th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>x2</td>
<td></td>
<td></td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>x3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xm</td>
<td>1</td>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Matching Problem: Structure Matching

Structures

<table>
<thead>
<tr>
<th></th>
<th>x1</th>
<th>y1</th>
<th></th>
<th>y2</th>
<th></th>
<th>y3</th>
<th></th>
<th>yn</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td>4</td>
<td></td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>x1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x2</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x3</td>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xm</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Deep Matching Models
Collaborators

Zhengdong Lu

Baotian Hu

Mingxuan Wang

Qun Liu

Qingcai Chen
Model: Deep Match Topic
Model: Deep Match Topic
(Lu & Li, NIPS 2013)

• Taking pairs of texts as input
• Learning topics of words in different granularities using LDA
• Taking topics in different granularities as hidden layers
• Constructing neural network using heuristics
• Learning parameters of neural network using back-propagation
Representing Posts and Comments as Bags of Words

Words in post and comment are viewed as different words.
Constructing Topics Using Latent Dirichlet Allocation

Bag of words

P_mats, P_catch, P_mice, C_great, C_mats

P_mats, P_chase, P_mice, C_poor, C_mice

Topic

P_mats
P_mice
C_mice
C_mats

P_chase
P_catch
C_great
C_poor

Topics in different granularities form different hidden layers
Construct Neural Network Using Heuristics

Four paired words in pink are connected to a hidden node

Four paired words in blue are connected to another hidden node
Architecture of Deep Matching Model

Examples
Local Model 1: (特产, 土产, 味道, ...) || (豆腐, 烤鸭, 甜, 野味, 糯米...)
Local Model 2: (路程, 安排, 地点, ...) || (距离, 安全, 隧道, 高速, 机票...)

烤鸭啊，想吃热乎的去烤鸭店，如全聚德，真空包装的超市就有
Experimental Results

• Data: 12,000 post-comment pairs from Weibo
• Cross validation in terms of P@1
• Conclusion: Deep Match works better than linear model

<table>
<thead>
<tr>
<th>Model</th>
<th>MAP</th>
<th>P@1</th>
</tr>
</thead>
<tbody>
<tr>
<td>P2R</td>
<td>0.565</td>
<td>0.489</td>
</tr>
<tr>
<td>P2R + P2P</td>
<td>0.621</td>
<td>0.567</td>
</tr>
<tr>
<td>P2R + MATCH</td>
<td>0.575</td>
<td>0.513</td>
</tr>
<tr>
<td>P2R + P2P + MATCH</td>
<td>0.621</td>
<td>0.574</td>
</tr>
</tbody>
</table>
Model: Deep Match Tree
Model: Deep Match Tree
(Wang, Lu, Li, & Liu; to appear)

- Taking pairs of dependency trees as input
- Mining frequent matching patterns from pairs of dependency trees
- Taking matching patterns as input layer of neural network
- Training weights of neural network using back propagation
Representing Sentence with Its Dependency Tree

Lexical and syntactic information of sentence is represented in its dependency tree.

马上从北京去纽约旅游喽
I will soon go from Beijing to New York for traveling

一定要到时代广场哦
You must visit the Times Square

应该去长城看看
You should go to see the Great Wall
Constructing Product of Trees

- Use two dependency-trees to create product of graph
- Represent interaction between two sentences
- A sub-graph represents a matching pattern
- Find high frequency patterns using pattern mining technique
Large Scale Graph Mining

- Large-scale graph mining for finding high frequency sub-graphs (matching patterns)
- Lexical and syntactic information is incorporated in the patterns

<table>
<thead>
<tr>
<th>Patterns without abstractions</th>
</tr>
</thead>
<tbody>
<tr>
<td>exam ⊗ score</td>
</tr>
<tr>
<td>Information theory ⊗ Shannon</td>
</tr>
<tr>
<td>thank→present ⊗ happy→birthday</td>
</tr>
<tr>
<td>win→game ⊗ trying→keep</td>
</tr>
<tr>
<td>out-of-control→prices ⊗ regulation</td>
</tr>
<tr>
<td>work→weekend ⊗ rest</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Patterns without abstractions</th>
</tr>
</thead>
<tbody>
<tr>
<td>hope→win→x ⊗ support→x</td>
</tr>
<tr>
<td>how about→x ⊗ like→x</td>
</tr>
<tr>
<td>gift→x ⊗ happy→x</td>
</tr>
<tr>
<td>recommend→x ⊗ x→nice</td>
</tr>
<tr>
<td>pretty good→x ⊗ fine→also→x'</td>
</tr>
</tbody>
</table>
Deep Match Tree

- Constructing deep neural network, with first layer corresponding mined patterns
Experimental Results

- Retrieval-based Conversation
- 5 million post-comment pairs for mining of patterns
- Data: 12,000 labeled post-comment pairs from Weibo
- Cross validation in terms of P@1
- Conclusion: Deep Match Tree outperforms baselines

<table>
<thead>
<tr>
<th>Model</th>
<th>P@1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.574</td>
</tr>
<tr>
<td>+DeepMatch</td>
<td>0.587</td>
</tr>
<tr>
<td>+WordEmbed</td>
<td>0.579</td>
</tr>
<tr>
<td>+Translation</td>
<td>0.585</td>
</tr>
<tr>
<td>+DeepMatch_tree</td>
<td>0.608</td>
</tr>
</tbody>
</table>
Model: CNN Match
Model: CNN Match
(Hu, Lu, Li, & Chen; NIPS 2014)

- Taking pairs of sentences as input
- Representing content of sentences and matching of sentences using Convolutional Neural Network
- No linguistic knowledge is needed
Sentence Model Using Convolutional Neural Network (CNN)

• Representing content of sentence using CNN
Advantage of Using CNN

- **Sliding windows**: possible sequences of words for composition
- **Convolution**: composition of words
- **Pooling**: selection of word sequences for composition
CNN Match Arc I:

• First represent two sentences, and then match
CNN Match Arc II:

• Represent and match two sentences simultaneously
• Two dimensional convolution and pooling
Advantages of CNN Match

• Order of words in two sentences are considered in the model
• Structures of two sentences can be captured
• Shared parameters improve efficiency in training
• Arc-II takes Arc-I as special case
Experimental Results

• Retrieval-based conversation
• Training data: 4 million pairs
• Testing data: 450k pairs
• Conclusion: Arc-II > Arc-I > baselines

<table>
<thead>
<tr>
<th>Model</th>
<th>P@1(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Guess</td>
<td>20.00</td>
</tr>
<tr>
<td>DEEPMATCH</td>
<td>49.85</td>
</tr>
<tr>
<td>WORDEMBED</td>
<td>54.31</td>
</tr>
<tr>
<td>SENMLP</td>
<td>52.22</td>
</tr>
<tr>
<td>SENNA+MLP</td>
<td>56.48</td>
</tr>
<tr>
<td>Arc-I</td>
<td>59.18</td>
</tr>
<tr>
<td>Arc-II</td>
<td>61.95</td>
</tr>
</tbody>
</table>
Summary
Summary

• Noah’s Ark Lab is working on intelligent information assistant
• Short Text Conversation (STC) is challenging yet interesting task
• Current approach = retrieval-based STC
• Learning to match is fundamental problem
• Big data and deep learning are powerful tools
• Several deep matching models proposed
Thank You!

hangli.hl@huawei.com