Learning to Ask Questions in Open-domain Conversational Systems with **Typed Decoders**

Yansen Wang$^1$, Chenyi Liu$^1$, **Minlie Huang$^1$$^,$ Liqiang Nie$^2$

$^1$Tsinghua University, China  
$^2$Shandong University, China

aihuang@tsinghua.edu.cn  
http://coai.cs.tsinghua.edu.cn/hml
Outline

- Background
- Problem & task definition
- Motivation
- Related works
- Method
- Experiments
- Summary
Good Examples from DL Models

User: 时间啊，你这把杀猪刀。。。 Time is a killer...

Machine: 我是被杀的 I am the one who was killed.

User: 最幸福的人不是拿到了最好的东西，而是学会去珍惜手上已经拥有的人。 The happiest person is not the one who gets the best thing, instead who cherishes what she/he already has.

Machine: 是的，所以我也要珍惜自己的幸福 Agree, I will cherish my happiness too.

User: 怕孤独，又舍不下自己的小自由。 Afraid to be lonely, but won’t give up my little freedom.

Machine: 我想去流浪，但没有人陪我去。 I want to wander, but no one accompany me.
Bad Examples are More!

User: 其实，他没有那么爱你。
   Ohh, he does not love you that much.
Machine: 幸福的一天开始了。
   A happy day starts.

User: 回北京了！很冷很高兴！
   Back to Beijing now. So cold so happy!
Machine: 什么时候来北京啊？
   When will you come to Beijing?

User: 会做饭的男人是最性感的男人。
   Mans who can cook are the most sexy.
Machine: 会做饭的男人是最性感的女人
   Mans who can cook are the most sexy women.
Bad Examples are More!

语义理解的问题 Semantic understanding

User: 你家多大？ How large is your house?
小冰: 这么快就要爆年龄了?
So soon to reveal my age?

User: 你是素食主义者还是吃肉啊？
Are you vegan or meatatarian?
小冰: 怎么减肥
How can I lose weight?

上下文理解的问题 Context issue

User: 早饭不好吃
小冰: 那明早一起吃
User: 行，你推荐的应该不错
小冰: 是吗？
User: 哈哈，我相信你
小冰: 呃呃

Inconsistency in personality

User: 你几岁了？
小冰: 我啊。。。反正是九零后！
User: 那你是几年出生的？
小冰: 错，75年阳历一月出生。
User: 你是九零后，怎么会75年出生？
小冰: 生在九零后，在深圳只能被当做八零后了。
Bad Examples (AI Ethics)

Picture from Prof. Frank Rudzicz, University of Toronto
Challenges in Chatting Machines

Semantics

Consistency

Interactiveness

Content, Context, Scene

Personality, Personalization, Language Style

Emotion & Sentiment

Strategy & Behavior
More Intelligent Chatting Machines

- Behaving more **interactively:**
  - Emotional Chatting Machine (AAAI 2018)
  - Proactive Behavior by Asking Good Questions (ACL 2018)
  - Controlling sentence function (ACL 2018)

- Behaving more **consistently:**
  - Explicit Personality Assignment (IJCAI-ECAI 2018)

- Behaving more **intelligently** with **semantics:**
  - Better Understanding and Generation Using Commonsense Knowledge (IJCAI-ECAI 2018 Distinguished Paper)

References:
2. Assigning personality/identity to a chatting machine for coherent conversation generation. **IJCAI-ECAI 2018.**
3. Commonsense Knowledge Aware Conversation Generation with Graph Attention. **IJCAI-ECAI 2018.**
4. Learning to Ask Questions in Open-domain Conversational Systems with Typed Decoders. **ACL 2018.**
5. Generating Informative Responses with Controlled Sentence Function. **ACL 2018.**
Problem & Task Definition

• How to ask **good** questions in open-domain conversational systems?

用户：我昨天晚上去聚餐了

Post: I went to dinner yesterday night.
Problem & Task Definition

用户：我昨天晚上去聚餐了
Post: I went to dinner yesterday night.

- Who were you with?
- Where did you have the dinner?
- How about the food?
- How many friends?
- Who paid the bill?
- Is it an Italian restaurant?
Problem & Task Definition

Scene: Dining at a restaurant

- Asking **good** questions requires **scene understanding**
Motivation

• Responding + asking (Li et al., 2016)
  • More interactive chatting machines

• Key proactive behaviors (Yu et al., 2016)
  • Less dialogue breakdowns

• Asking good questions is indication of understanding
  • As in course teaching
  • Scene understanding in this paper
Related Work

• Traditional question generation (Andrenucci and Sneiders, 2005; Popowich and Winne, 2013)
• Syntactic Transformation

• **Given context:** As recently as 12,500 years ago, the Earth was in the midst of a glacial age referred to as the Last Ice Age.
• **Generated question:** How would you describe the Last Ice Age?
Related Work

- A few neural models for question generation in reading comprehension (Du et al., 2017; Zhou et al., 2017; Yuan et al., 2017)

Given

- **Passage**: ...Oxygen is used in cellular respiration and released by *photosynthesis*, which uses the energy of sunlight to produce oxygen from water. ...
- **Answer**: photosynthesis
- **Generated question**: What life process produces oxygen in the presence of light?
Related Work

• Visual question generation for **eliciting interactions** (Mostafazadeh, 2016): beyond image captioning

• **Given image:**

• **Generated question:** What happened?
Difference to Existing Works

• Different goals:
  • To enhance **interactiveness and persistence** of human-machine interactions
  • **Information seeking** in read comprehension

• Various patterns: YES-NO, WH-, HOW-ABOUT, etc.

• **Topic transition**: from topics in post to topics in response
  • Dinner → food; fat → climbing; sports → soccer
Key Observations

- A good question is a natural **composition** of
  - **Interrogatives** for using various questioning patterns
  - **Topic words** for addressing interesting yet novel topics
  - **Ordinary words** for playing grammar or syntactic roles

Example 1:
User: I am too *fat* ...
Machine: **How about** *climbing* this weekend?

Example 2:
User: Last night, I stayed in *KTV* with friends.
Machine: **Are you happy with your** *singing*?
Hard/Soft Typed Decoders
(HTD/STD)
Encoder-decoder Framework

Encoder:
post: The cake tastes good <EOS>

Decoder:
response: Is it a cheese cake

\[
X = x_1 x_2 \cdots x_m \\
Y = y_1 y_2 \cdots y_n \\
Y^* = \text{argmax } \mathcal{P}(Y|X). \\
\]

\[
\mathcal{P}(y_t|y_{<t}, X) = \text{MLP}(s_t, e(y_{t-1}), c_t), \\
s_t = \text{GRU}(s_{t-1}, e(y_{t-1}), c_t), \\
c_t = \sum_{i=1}^{T} \alpha_{t,i} h_i \\
h_t = \text{GRU}(h_{t-1}, e(x_t)),
\]
Soft Typed Decoder (STD)

Encoder:

post: The cake tastes good <EOS>

Decoder:

response: Is it a cheese cake

Soft Typed Decoder (STD)

Attentive Read

Decoding state

Decoder:

<EOS> Is it a cheese cake

0.5 0.3 0.2

0.5 0.3 0.2

type I type II type III
type I type II type III
type specific generation distributions

final generation distribution

mixture

type distribution

vocab.

vocab.

vocab.
Soft Typed Decoder (STD)

- Applying **multiple type-specific generation distributions** over the same vocabulary

- Each word has a **latent** distribution among the set \text{type}(w) \in \{\text{interrogative, topic word, ordinary word}\}

- STD is a very simple **mixture** model

\[
P(y_t | y_{<t}, X) = \sum_{i=1}^{k} P(y_t | ty_t = c_i, y_{<t}, X) \cdot P(ty_t = c_i | y_{<t}, X),
\]
Soft Typed Decoder (STD)

• Estimate the type distribution of each word:

\[ P(ty_t|y_t, X) = \text{softmax}(W_0 s_t + b_0), \]

• Estimate the type-specific generation distribution of each word:

\[ P(y_t|ty_t = c_i, y_t, X) = \text{softmax}(W_{c_i} s_t + b_{c_i}), \]

• The final generation distribution is a mixture of the three type-specific generation distribution.

\[ P(y_t|y_t, X) = \sum_{i=1}^{k} P(y_t|ty_t = c_i, y_t, X) \cdot P(ty_t = c_i|y_t, X), \]
Hard Typed Decoder (HTD)

• In soft typed decoder, word types are modeled in a latent, implicit way

• Can we control the word type more explicitly in generation?
  • Stronger control
Hard Typed Decoder (HTD)

Encoder:
post: The cake tastes good \(<\text{EOS}>\)

Decoder:
\(<\text{EOS}>\), is, it, a, cheese

Response: Is it a cheese cake

Hard Typed Decoder (HTD)

Gumbel-softmax

<table>
<thead>
<tr>
<th>Type</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type I</td>
<td>0.9</td>
</tr>
<tr>
<td>Type II</td>
<td>0.07</td>
</tr>
<tr>
<td>Type III</td>
<td>0.03</td>
</tr>
</tbody>
</table>

Vocab.

Type prob. distribution:
- Type I: 0.5
- Type II: 0.3
- Type III: 0.2

Generation prob. distribution

Decoding state

Final probability

Type I

Type II

Type III

Cake
Hard Typed Decoder (HTD)

- Estimate the generation probability distribution
  \[ P(y_t | y_{<t}, X) = \text{softmax}(W_0 s_t + b_0). \]
- Estimate the type probability distribution
  \[ P(ty_t | y_{<t}, X) = \text{softmax}(W_1 s_t + b_1). \]
- Modulate words’ probability by its corresponding type probability:
  \[ P'(y_t | y_{<t}, X) = P(y_t | y_{<t}, X) \cdot m(y_t), \]

\(m(y_t)\) is related to the type probability of word \(y_t\)
**Hard Typed Decoder (HTD)**

<table>
<thead>
<tr>
<th>Generation distr.</th>
<th>Type distr.</th>
<th>Modulated distr.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>what</strong> 0.3</td>
<td>$T_{\text{interrogative}}$ 0.7</td>
<td><strong>what</strong> 0.8</td>
</tr>
<tr>
<td><strong>food</strong> 0.2</td>
<td>$T_{\text{topic}}$ 0.1</td>
<td>$\rightarrow$ <strong>food</strong> 0.05</td>
</tr>
<tr>
<td><strong>is</strong> 0.4</td>
<td>$T_{\text{ordinary}}$ 0.2</td>
<td><strong>is</strong> 0.09</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- **Argmax?** (firstly select largest type prob. then sample word from generation dist.)
- Indifferentiable
- Serious grammar errors if word type is wrongly selected
Hard Typed Decoder (HTD)

- **Gumble-Softmax:**
  - A differentiable surrogate to the \texttt{argmax} function.

\[
m(y_t) = \text{GS}(\mathcal{P}(ty_t = c(y_t) | y_{<t}, X)),
\]

\[
\text{GS}(\pi_i) = \frac{e^{(\log(\pi_i) + g_i)/\tau}}{\sum_{j=1}^{k} e^{(\log(\pi_j) + g_j)/\tau}},
\]

<table>
<thead>
<tr>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type I</td>
<td>0.9</td>
</tr>
<tr>
<td>Type II</td>
<td>0.07</td>
</tr>
<tr>
<td>Type III</td>
<td>0.03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type I</td>
<td>0.5</td>
</tr>
<tr>
<td>Type II</td>
<td>0.3</td>
</tr>
<tr>
<td>Type III</td>
<td>0.2</td>
</tr>
</tbody>
</table>
Hard Typed Decoder (HTD)

- In HTD, the types of words are given in advance.
- *How to determine the word types?*
Hard Typed Decoder (HTD)

• Interrogatives:
  • A list of about 20 interrogatives are given by hand.
• Topic words:
  • Training: all nouns and verbs in response are topic words.
  • Test: 20 words are predicted by PMI.

\[
PMI(w_x, w_y) = \log \frac{p(w_x, w_y)}{p_1(w_x) \cdot p_2(w_y)},
\]

\[
Rel(k_i, X) = \sum_{w_x \in X} e^{PMI(w_x, k_i)},
\]

• Ordinary words:
  • All other words, for grammar or syntactic roles
Loss Function

- Cross entropy
- Supervisions are on both final probability and the type distribution:

\[ \Phi_1 = \sum_t - \log P(y_t = \tilde{y}_t | y_{<t}, X), \]
\[ \Phi_2 = \sum_t - \log P(ty_t = \tilde{y}_t | y_{<t}, X), \]
\[ \Phi = \Phi_1 + \lambda \Phi_2, \]

- \( \lambda \) is a term to balance the two kinds of losses.
Experiments
Dataset

- PMI estimation: calculated from 9 million post-response pairs from Weibo.
- Dialogue Question Generation Dataset (DQG), about 491,000 pairs:
  - Distilled questioning responses using about 20 hand-draft templates
  - Removed universal questions
  - Available at http://coai.cs.tsinghua.edu.cn/hml/dataset/
Baselines

• **Seq2Seq**: A simple encoder-decoder model ([Luong et al., 2015](#))

• **Mechanism-Aware (MA)**: Multiple responding mechanisms represented by real-valued vectors ([Zhou et al., 2017](#))

• **Topic-Aware (TA)**: Topic Aware Model by incorporating topic words ([Xing et al., 2017](#))

• **Elastic Responding Machine (ERM)**: Enhanced MA using reinforcement learning ([Zhou et al., 2018](#))
Automatic Evaluation

<table>
<thead>
<tr>
<th>Model</th>
<th>Perplexity</th>
<th>Distinct-1</th>
<th>Distinct-2</th>
<th>TRR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq2Seq</td>
<td>63.71</td>
<td>0.0573</td>
<td>0.0836</td>
<td>6.6%</td>
</tr>
<tr>
<td>MA</td>
<td><strong>54.26</strong></td>
<td>0.0576</td>
<td>0.0644</td>
<td>4.5%</td>
</tr>
<tr>
<td>TA</td>
<td>58.89</td>
<td>0.1292</td>
<td>0.1781</td>
<td>8.7%</td>
</tr>
<tr>
<td>ERM</td>
<td>67.62</td>
<td>0.0355</td>
<td>0.0710</td>
<td>4.5%</td>
</tr>
<tr>
<td>STD</td>
<td>56.77</td>
<td>0.1325</td>
<td>0.2509</td>
<td>12.1%</td>
</tr>
<tr>
<td>HTD</td>
<td>56.10</td>
<td><strong>0.1875</strong></td>
<td><strong>0.3576</strong></td>
<td><strong>43.6%</strong></td>
</tr>
</tbody>
</table>

Table 1: Results of automatic evaluation.

**Evaluation metrics**

- **Perplexity & Distinct**
- **TRR (Topical Response Ratio):**
  - 20 topic words are predicted with PMI for each post.
  - TRR is the proportion of the responses containing at least one topic word.
Manual Evaluation

• Pair-wise comparison: win, loss, tie
• Three evaluation criteria:
  • Appropriateness: whether a question is reasonable in logic and content, and has key info.
  • Richness: containing topic words or not
  • Willingness to respond to a generated question
## Manual Evaluation (Pairwise)

<table>
<thead>
<tr>
<th>Models</th>
<th>Appropriateness</th>
<th>Richness</th>
<th>Willingness</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Win (%)</td>
<td>Lose (%)</td>
<td>Tie (%)</td>
</tr>
<tr>
<td>STD vs. Seq2Seq</td>
<td>42.0</td>
<td>38.6</td>
<td>19.4</td>
</tr>
<tr>
<td>STD vs. MA</td>
<td>39.6*</td>
<td>31.2</td>
<td>29.2</td>
</tr>
<tr>
<td>STD vs. TA</td>
<td>42.2</td>
<td>40.0</td>
<td>17.8</td>
</tr>
<tr>
<td>STD vs. ERM</td>
<td>43.4*</td>
<td>34.4</td>
<td>22.2</td>
</tr>
<tr>
<td>HTD vs. Seq2Seq</td>
<td>50.6**</td>
<td>30.6</td>
<td>18.8</td>
</tr>
<tr>
<td>HTD vs. MA</td>
<td>54.8**</td>
<td>24.4</td>
<td>20.8</td>
</tr>
<tr>
<td>HTD vs. TA</td>
<td>52.0**</td>
<td>38.2</td>
<td>9.8</td>
</tr>
<tr>
<td>HTD vs. ERM</td>
<td>64.8**</td>
<td>23.2</td>
<td>12.0</td>
</tr>
<tr>
<td>HTD vs. STD</td>
<td>52.0**</td>
<td>33.0</td>
<td>15.0</td>
</tr>
</tbody>
</table>

Table 2: Annotation results. Win for “A vs. B” means A is better than B. Significance tests with Z-test were conducted. Values marked with * means $p$-value $< 0.05$, and ** for $p$-value $< 0.01$.

**Score:** the probability of win/lose/tie of our model vs. baseline
### Examples

| Post: | 我们昨天去唱歌了
We went to karaoke yesterday. |
|-------|-------------------------------|
| Seq2Seq: | 什么时候?
When? |
| MA: | 你怎么知道我是在问
How did you know I am questioning you? |
| TA: | 什么?
What? |
| ERM: | 什么情况?
What happened? |
| STD: | 去哪里唱歌?
Where did you sing karaoke? |
| HTD: | 你们几个人唱歌?
How many people were singing with you? |

Table 4: Generated question examples. Interrogative words are bolded and topic words underlined.
### More Examples

- **Different questioning patterns and topic transition:**

| Post-3: 师在体育中心上班。 | I work at the sports center. |
| Human: 你每天从家过去要多久？ | How long does it take you to go there from home? |
| STD: 你是什么部门的啊? | Which department are you from? |
| HTD: 那你是体育学院的学生吗? | Are you a student from the College of Physical Education? |

| Post-4: 今天的寿司真好吃。 | The sushi today is so delicious! |
| Human: 我记得你不喜欢吃日式料理的不是吗? | I remember you didn’t like eating Japanese food, did you? |
| STD: 你什么时候来无锡，我请你。 | When will you come to Wuxi? It is my treat. |
| HTD: 是吗?改天尝尝! | Really? I will try sometime! |
### Visualization

- Type prediction at each decoding position

<table>
<thead>
<tr>
<th>Decoding steps</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Post:</strong> 我喜欢小动物(I like little animals)</td>
<td><strong>Response:</strong> 你(you)</td>
<td>喜欢(like)</td>
<td>兔子(rabbit)</td>
<td>吗(particle)</td>
<td>?</td>
<td>_EOS</td>
</tr>
<tr>
<td>Interrogative</td>
<td>0.09</td>
<td>0.02</td>
<td>0.01</td>
<td>0.85</td>
<td>1.00</td>
<td>0.01</td>
</tr>
<tr>
<td>Topic word</td>
<td>0.26</td>
<td>0.35</td>
<td>0.71</td>
<td>0.14</td>
<td>0.00</td>
<td>0.02</td>
</tr>
<tr>
<td>Ordinary word</td>
<td>0.65</td>
<td>0.63</td>
<td>0.28</td>
<td>0.01</td>
<td>0.00</td>
<td>0.97</td>
</tr>
</tbody>
</table>
Summary

• Stronger control in language generation via word semantic type
• What’s new
  • A new task: question generation in open-domain dialogue systems
  • A new dataset: Dialog Question Generation Dataset
  • A new model with two variants: possibly applicable to other generation tasks if word semantic types can be easily identified
• The compatibility issue between topic control and other word type control is NOT well solved
  • Bad grammar or not reasonable responses
Thanks for your attentions

- Codes: [https://github.com/victorywys/Learning2Ask_TypedDecoder](https://github.com/victorywys/Learning2Ask_TypedDecoder)
- Homepage: [http://coai.cs.tsinghua.edu.cn/hml](http://coai.cs.tsinghua.edu.cn/hml)
- Recruiting post-doctors!
Error Analysis

• Main error types
  • *No topic words (NoT)* in a response
  • *Wrong topics (WrT)* where topic words are irrelevant
  • *Type generation error (TGE)* where a wrong word type is predicted

<table>
<thead>
<tr>
<th>Error Type</th>
<th>NoT</th>
<th>WrT</th>
<th>TGE</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>STD</td>
<td>34%</td>
<td>34%</td>
<td>29%</td>
<td>3%</td>
</tr>
<tr>
<td>HTD</td>
<td>29%</td>
<td>39%</td>
<td>29%</td>
<td>3%</td>
</tr>
</tbody>
</table>

Table 6: Error type distribution.
Error Analysis: Examples

No topic words

Post-1: 今天好开心啊!
I am so happy today!
STD: 你怎么知道?
How do you know?

Wrong topics

Post-2: 海报非常棒，期待若曦与我们男人的首度合作。
The poster is great and we look forward to our first cooperation with Ruoxi.
HTD: 你海报怎么样啊?
How about your poster?

Type generation error

Post-3: 又生病啦?吃点药就好了。
Got sick again? Just take some medicine and you’ll be fine soon.
STD: 我也不知道怎么回事。
I don’t know what happened.
HTD: 肠胃痛了，上火吗?
Stomach aching, ulcer?