Commonsense Reasoning using Pre-trained Language Models
Today: Language models + commonsense reasoning

Outline

- Commonsense reasoning
- Pre-trained language models
- The four ways of using PLTM for commonsense reasoning:
  1. Pre-training
  2. Retrieval-based augmentation
  3. Model-based augmentation
  4. Formal logic and symbolic reasoning
Commonsense reasoning
Commonsense reasoning

Definition

- Basic level of **practical knowledge and reasoning** concerning **everyday situations** and events that are commonly **shared among most** people [1].

- Examples:
  - Okay to keep the **closet door open**, but not the **fridge door open**
  - More rain **causes** more greenery
  - If you **give someone a nice gift** they will be **happy**

Commonsense reasoning

Applications

- Basic level of practical knowledge and reasoning concerning everyday situations and events that are commonly shared among most people.

- Popular downstream tasks
  - Question answering
  - Generation (e.g., graph generation for interpretability)

- Grand goal
  - Build machines that can reason about the world like humans do
## Commonsense reasoning

### Task-oriented definition

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Train</th>
<th>Development</th>
<th>Test</th>
<th>Source Example</th>
<th>Target Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommonsenseQA</td>
<td>9,741</td>
<td>1,221</td>
<td>1,140</td>
<td>context: <em>What home entertainment equipment requires cable?</em> options: 1: <em>radio shack</em> 2: <em>substation</em> 3: <em>cabinet</em> 4: <em>television</em> 5: <em>desk</em></td>
<td>4</td>
</tr>
<tr>
<td>OpenbookQA</td>
<td>4,957</td>
<td>500</td>
<td>500</td>
<td>context: <em>You can make a telescope with</em> options: 1: <em>straw</em> 2: <em>glass</em> 3: <em>candle</em> 4: <em>mailing tube</em></td>
<td>2</td>
</tr>
<tr>
<td>PIQA</td>
<td>16,113</td>
<td>1,838</td>
<td>3,084</td>
<td>context: <em>When boiling butter, when it's ready, you can</em> options: 1: <em>Pour it onto a plate</em> 2: <em>Pour it into a jar</em></td>
<td>2</td>
</tr>
<tr>
<td>aNLI</td>
<td>169,654</td>
<td>1,532</td>
<td>3,040</td>
<td>context: <em>It was my birthday. When I got home the party was set up for my brother.</em> options: 1: <em>I was so excited.</em> 2: <em>I was so mad.</em></td>
<td>2</td>
</tr>
<tr>
<td>CommonGEN</td>
<td>67,389</td>
<td>4,018</td>
<td>6,042</td>
<td>generate a sentence with these concepts: <em>Apple Grow Tree</em></td>
<td><em>Apple grows on the tree</em></td>
</tr>
</tbody>
</table>

Zhou, Wangchunshu, Dong-Ho Lee, Ravi Kiran Selvam, Seyeon Lee, Bill Yuchen Lin, and Xiang Ren. "Pre-training text-to-text transformers for concept-centric common sense." *ICLR 2021*
Defeasible Reasoning

- A classification task
- Given a premise $P$, a hypothesis $H$
  - New evidence (update) $U$ may be weaken or strengthen the hypothesis


How to best use graphs for defeasible reasoning? (We’ll discuss in detail later)

Given a defeasible query $PHU$

And a graph generated for the query (augmented information)

Generate classification label
Commonsense reasoning

Recent trend

True number likely much higher: *commonsense reasoning* is not always explicitly mentioned
Today: Language models + commonsense reasoning

Outline

- Commonsense reasoning ✓
- Pre-trained language models
- The four ways of using PLTM for commonsense reasoning:
  1. Pre-training
  2. Retrieval-based augmentation
  3. Model-based augmentation
  4. Formal logic and symbolic reasoning
Pre-trained language models
Pre-trained language models

TL; DR

- Pre-trained language models:
  - Transformers based deep neural networks
  - Trained on web-scale text corpora
  - Goal is to learn **informative** representations of text

- Language Models
  - Contextualized token embedding: BERT, XL-Net, Roberta,
  - Next-token Prediction: GPT-N
  - Hybrid: BART, T5
Pre-trained language models

Tasks

- Generative tasks (Sequence-to-Sequence Tasks)
  - Machine Translation: English sentence → Chinese sentence
  - Text Summarization: News document → Summary
  - Graph generation: Context → Event Graph

- Discriminative tasks
  - Multi-choice question-answering
  - Answer-span generation
  - Ranking
Pre-trained language models
Pre-training + fine-tuning

- The defacto way of approaching most NLP tasks currently

- Requires:
  - A dataset with samples \((X, y)\)

- Two steps
  - Start from a pre-trained model \(M\) (e.g., BART)
  - Fine-tune \(M\) to perform better on \(X \rightarrow y\)

- Intuition:
  - Pre-training imparts the model with knowledge of the language
Language models are getting huge + impressive

- Diminishing returns in training the model.
  - Practically impossible
  - The largest model has 530B parameters
  - Practical applications of language generation near:
    - https://copilot.github.com/
    - Potentially disruptive
  - Put these two things together:
    - LLM are a fact of life now (or will be soon).
    - New methods to make the best use of them

Using DeepSpeed and Megatron to Train
Megatron-Turing NLG 530B, the World's Largest
and Most Powerful Generative Language Model

Today: Language models + commonsense reasoning

Outline

- Commonsense reasoning ✔
- Pre-trained language models ✔
- The four ways of using PLTM for commonsense reasoning:
  1. Pre-training
  2. Retrieval-based augmentation
  3. Model-based augmentation
  4. Formal logic and symbolic reasoning
Pre-training strategies for commonsense reasoning
Published as a conference paper at ICLR 2021

PRE-TRAINING TEXT-TO-TEXT TRANSFORMERS FOR CONCEPT-CENTRIC COMMON SENSE

Wangchunshu Zhou\textsuperscript{1,*}, Dong-Ho Lee\textsuperscript{2,*}, Ravi Kiran Selvam\textsuperscript{2}, Seyeon Lee\textsuperscript{2}, Bill Yuchen Lin\textsuperscript{2}, Xiang Ren\textsuperscript{2}
\textsuperscript{1}Beihang University, \textsuperscript{2}University of Southern California
zhouwangchunshu@buaa.edu.cn, \{dongho.lee, xiangren\}@usc.edu
Pre-training text-to-text transformers

Overview

- Perform additional pre-training on top of an existing language model
- Add three self-supervised tasks that are more useful for commonsense reasoning
- Two generative tasks:
  - Concept-to-sentence
  - Concept order recovery
- One discriminative task
  - Distinguish between sentence that follows commonsense and one that does not
Pre-training text-to-text transformers

Generative task

- Self-supervised: does not require labels (but requires special annotations)

Concept-to-Sentence

Input: $<c2s>$ Generate a sentence with the concepts: forward, Simpson, ignore, information, prosecutor

Output: The information was forwarded to Simpson's prosecutors, but it was ignored.

Concept Order Recovering

Input: $<cor>$ Correct the order of the given sentence: Rahul stops him, fights his bar, and drives to a local performance.

Output: Rahul fights him, stops his performance, and drives to a local bar.

$$L_{c2s} = E \left( \sum_{i=1}^{n} - \log p(x_i|<c2s>; \text{PERMUTE}(C); x_{1:i-1}) \right)$$

$$L_{cor} = E \left( \sum_{i=1}^{n} - \log p(x_i|<cor>; \text{CONCEPT-PERMUTE}(x,C); x_{1:i-1}) \right)$$
Pre-training text-to-text transformers

Discriminative task

- Distinguish between real and fake

**Generative QA**

Input: \(<\text{cont}>\) Which sentence is correct?: options:
1. The increased \textbf{number} of male visitors \textbf{inspired} by the \textbf{article} \textbf{raised} security concerns
2. The increased \textbf{article} of male visitors \textbf{raised} by the \textbf{number} \textbf{inspired} security concerns

Output: The increased \textbf{number} of male visitors \textbf{inspired} by the \textbf{article} \textbf{raised} security concerns
Pre-training text-to-text transformers

Joint training

- First train individually on both the tasks, then do another round of joint training

\[
L_{\text{cont}_\text{joint}_\text{c2s}} = E \left( - \log D_{\phi}(y|<\text{cont}>; x; G_{\theta}(<\text{c2s}>; \text{PERMUTE}(C))) \right)
\]

\[
L_{\text{cont}_\text{joint}_\text{cor}} = E \left( - \log D_{\phi}(y|<\text{cor}>; x; G_{\theta}(<\text{cor}>; \text{CONCEPT-PERMUTE}(x, C))) \right)
\]

\[
L_{\text{joint}} = (L_{\text{c2s}} + L_{\text{cor}}) + \beta (L_{\text{cont}_\text{joint}_\text{c2s}} + L_{\text{cont}_\text{joint}_\text{cor}})
\]
Pre-training text-to-text transformers

Experiments

- Pre-train on 500k sentences from wikipedia using the three objectives, and then fine-tune on individual tasks.

- Experiments on five commonsense datasets
## Pre-training text-to-text transformers

### Results

<table>
<thead>
<tr>
<th>Methods</th>
<th>CSQA</th>
<th>OBQA</th>
<th>PIQA</th>
<th>aNLI</th>
<th>BLEU-4</th>
<th>METEOR</th>
<th>CIDEr</th>
<th>SPICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT-base</td>
<td>53.08±0.16</td>
<td>57.60±0.8</td>
<td>64.86±0.52</td>
<td>61.88±0.56</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>ERNIE</td>
<td>54.06±0.12</td>
<td>58.90±0.9</td>
<td>66.47±0.58</td>
<td>63.04±0.46</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>KnowBERT</td>
<td>53.88±0.15</td>
<td>58.50±0.8</td>
<td>66.61±0.63</td>
<td>63.18±0.52</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>T5-base</td>
<td>61.88±0.08</td>
<td>58.20±1.0</td>
<td>68.14±0.73</td>
<td>61.10±0.38</td>
<td>24.90</td>
<td>31.20</td>
<td>12.99</td>
<td>32.40</td>
</tr>
<tr>
<td>T5-base + cont. pretraining</td>
<td>61.92±0.45</td>
<td>58.10±0.9</td>
<td>68.19±0.77</td>
<td>61.15±0.52</td>
<td>25.10</td>
<td>31.00</td>
<td>13.12</td>
<td>32.40</td>
</tr>
<tr>
<td>T5-base + SSM</td>
<td>62.08±0.41</td>
<td>58.30±0.8</td>
<td>68.27±0.71</td>
<td>61.25±0.51</td>
<td>25.20</td>
<td>31.20</td>
<td>13.28</td>
<td>32.40</td>
</tr>
<tr>
<td>CALM (Generative-Only)</td>
<td>62.28±0.36</td>
<td>58.90±0.4</td>
<td>68.91±0.88</td>
<td>60.95±0.46</td>
<td>25.80</td>
<td>31.20</td>
<td>13.81</td>
<td>32.60</td>
</tr>
<tr>
<td>CALM (Contrastive-Only)</td>
<td>62.73±0.41</td>
<td>59.30±0.3</td>
<td>70.67±0.98</td>
<td>61.35±0.06</td>
<td>25.50</td>
<td>31.20</td>
<td>13.58</td>
<td>32.60</td>
</tr>
<tr>
<td>CALM (w/o Mix warmup)</td>
<td>62.18±0.48</td>
<td>59.00±0.5</td>
<td>69.21±0.57</td>
<td>61.25±0.55</td>
<td>25.80</td>
<td>31.20</td>
<td>13.77</td>
<td>32.60</td>
</tr>
<tr>
<td>CALM (Mix-only)</td>
<td>63.02±0.47</td>
<td>60.40±0.4</td>
<td>70.07±0.98</td>
<td>62.79±0.55</td>
<td>26.00</td>
<td>31.20</td>
<td>13.82</td>
<td>32.80</td>
</tr>
<tr>
<td>CALM</td>
<td><strong>63.32±0.35</strong></td>
<td><strong>60.90±0.4</strong></td>
<td><strong>71.01±0.61</strong></td>
<td><strong>63.20±0.52</strong></td>
<td><strong>26.40</strong></td>
<td><strong>31.40</strong></td>
<td><strong>13.88</strong></td>
<td><strong>33.00</strong></td>
</tr>
</tbody>
</table>

- **Directly train with the joint objective**
- **Separately train the two objectives**

T5-base + 3 training objectives
Pre-training text-to-text transformers

Takeaways/questions

- What if T5-base is pre-trained on the **same data** without special objectives?

- Commonsense pre-training helps on downstream commonsense tasks

- Non-trivial, as common assumption is that vanilla pre-training is sufficient for commonsense reasoning

<table>
<thead>
<tr>
<th>method</th>
<th>#parameters</th>
<th>CSQA</th>
<th>OBQA</th>
<th>PIQA</th>
<th>aNLI</th>
</tr>
</thead>
<tbody>
<tr>
<td>T5-large</td>
<td>774M</td>
<td>69.81</td>
<td>61.40</td>
<td>72.19</td>
<td>75.54</td>
</tr>
<tr>
<td>CALM-large</td>
<td>774M</td>
<td>71.31</td>
<td>66.00</td>
<td>75.11</td>
<td>77.12</td>
</tr>
<tr>
<td>BERT-large</td>
<td>345M</td>
<td>57.06</td>
<td>60.04</td>
<td>67.08</td>
<td>66.75</td>
</tr>
<tr>
<td>RoBERTa-large</td>
<td>345M</td>
<td>71.81</td>
<td>63.90</td>
<td>76.90</td>
<td>82.35</td>
</tr>
<tr>
<td>SOTA</td>
<td>11B</td>
<td>79.1</td>
<td>87.2</td>
<td>90.13</td>
<td>89.70</td>
</tr>
</tbody>
</table>
Using novel pre-training objectives for commonsense reasoning

Additional references

- Towards Zero-shot Commonsense Reasoning with Self-supervised Refinement of Language Models

- Eigen: Event influence generation using pre-trained language models
Today: Language models + commonsense reasoning

Outline

- Commonsense reasoning ✔
- Pre-trained language models ✔
- The four ways of using PLTM for commonsense reasoning:
  1. Pre-training ✔
  2. Retrieval-based augmentation
  3. Model-based augmentation
  4. Formal logic and symbolic reasoning
Retrieval-based augmentation
Retrieval-based augmentation

Overview

- High-level idea:
  - Use the given commonsense question as a query to get more information from the web or knowledge bases (conceptnet/wikidata)

- Why:
  - Language models might not be able to leverage the context (especially the smaller language models)
  - Might be easier to find pin-pointed information from structured knowledge bases
  - Models are outdated, text on the web is constantly updated
Fusing Context Into Knowledge Graph for Commonsense Question Answering

Yichong Xu*, Chenguang Zhu*, Ruochen Xu, Yang Liu, Michael Zeng, Xuedong Huang
Microsoft Cognitive Services Research Group
{yicxu, chezhu, ruox, yaliu10, nzeng, xdh}@microsoft.com

ACL 2021
Overview

- Given a multiple choice commonsense question:
  - Identify entities in the question and choice
  - Identify triples from conceptnet that connect question and answer.
  - Use wiktionary to retrieve definition of all the concepts mention in the question and answer choices
  - Feed the question and the choices individually to ALBERT, and classify
Fusing Context Into Knowledge Graph for Commonsense Question Answering

Idea
Fusing Context Into Knowledge Graph for Commonsense Question Answering

Results

<table>
<thead>
<tr>
<th>Methods</th>
<th>Single</th>
<th>Ensemble</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT+OMCS</td>
<td>62.5</td>
<td>-</td>
</tr>
<tr>
<td>RoBERTa</td>
<td>72.1</td>
<td>72.5</td>
</tr>
<tr>
<td>RoBERTa+HyKAS</td>
<td>73.2</td>
<td>-</td>
</tr>
<tr>
<td>XLNet+DREAM</td>
<td>-</td>
<td>73.3</td>
</tr>
<tr>
<td>RoBERTa+KE</td>
<td>73.3</td>
<td>-</td>
</tr>
<tr>
<td>RoBERTa+KEDGN</td>
<td>-</td>
<td>74.4</td>
</tr>
<tr>
<td>XLNet+GraphReason</td>
<td>75.3</td>
<td>-</td>
</tr>
<tr>
<td>ALBERT</td>
<td>-</td>
<td>76.5</td>
</tr>
<tr>
<td>RoBERTa+MHGRN</td>
<td>75.4</td>
<td>76.5</td>
</tr>
<tr>
<td>ALBERT+PG-Full</td>
<td>75.6</td>
<td>78.2</td>
</tr>
<tr>
<td>T5</td>
<td>78.1</td>
<td>-</td>
</tr>
<tr>
<td>ALBERT+KRD</td>
<td>78.4</td>
<td>-</td>
</tr>
<tr>
<td>UnifiedQA</td>
<td>79.1</td>
<td>-</td>
</tr>
<tr>
<td>ALBERT+KCR</td>
<td>79.5</td>
<td>-</td>
</tr>
<tr>
<td>DEKCOR (ours)</td>
<td><strong>80.7</strong></td>
<td><strong>83.3</strong></td>
</tr>
</tbody>
</table>

OpenBook QA

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT + Careful Selection</td>
<td>72.0</td>
</tr>
<tr>
<td>AristoRoBERTa</td>
<td>77.8</td>
</tr>
<tr>
<td>ALBERT + KB</td>
<td>81.0</td>
</tr>
<tr>
<td>ALBERT + PG-Full</td>
<td>81.8</td>
</tr>
<tr>
<td>TTTTT (T5-3B)</td>
<td>83.2</td>
</tr>
<tr>
<td>UnifiedQA (T5-11B)</td>
<td><strong>87.2</strong></td>
</tr>
<tr>
<td>DEKCOR (ours)</td>
<td>82.4</td>
</tr>
</tbody>
</table>
Retrieval Enhanced Model for Commonsense Generation

Han Wang\textsuperscript{1*}, Yang Liu\textsuperscript{2}, Chenguang Zhu\textsuperscript{2}, Linjun Shou\textsuperscript{3}, Ming Gong\textsuperscript{3}, Yichong Xu\textsuperscript{2}, Michael Zeng\textsuperscript{2}

\textsuperscript{1}New York University
\textsuperscript{2}Microsoft Cognitive Services Research Group
\textsuperscript{3}STCA NLP Group, Microsoft, Beijing, China

hwang@nyu.edu
{yaliu10, chezhu, lisho, migon, yicxu, nzeng}@microsoft.com

ACL 2021
Retrieval Enhanced Model for Commonsense Generation

Overview

- Task: commongen
  - \{tree, apple, grow\} \rightarrow Apples grow on tree.

- Method:
  - For a given set of input concepts, retrieve sentences that contain them.
  - Re-rank the retrieved sentences.

- Also do CALM style pre-training
Retrieval Enhanced Model for Commonsense Generation

Method

- Trainable retriever: train BERT to rank the true sentence the highest (binary classification task).

\[ \text{score}(v) > \text{score}(z_i) \]
Retrieval Enhanced Model for Commonsense Generation

Example

<table>
<thead>
<tr>
<th>Concept Set:</th>
<th>trailer shirt side sit road</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>T5:</strong></td>
<td>A man sits on the side of a trailer and a shirt.</td>
</tr>
<tr>
<td><strong>Matching Retriever:</strong></td>
<td>(1) Two guys in red shirts are sitting on chairs, by the side of the road, behind that open trailer. (2) Two men, one wearing a straw cone hat, blue shirt, talking with a guy in a tan sunhat, red plaid shirt, both with baskets in front of them, sitting on the side of a dirt road. (3) An older guy with a tan shirt and hat sitting on the side of a road with bricks all around him and a small green bowl on the side.</td>
</tr>
<tr>
<td><strong>RE-T5(matching retriever):</strong></td>
<td>a man in a tan shirt sits on the side of a road.</td>
</tr>
<tr>
<td><strong>Trainable Retriever:</strong></td>
<td>(1) Two guys in red shirts are sitting on chairs, by the side of the road, behind that open trailer. (2) Teenagers in matching shirts stand at the side of the road holding trash bags. (3) A man in a white shirt and black pants standing at the side or the road.</td>
</tr>
<tr>
<td><strong>RE-T5(trainable retriever):</strong></td>
<td>a man in a white shirt and black pants sits on the side of a trailer on the road.</td>
</tr>
</tbody>
</table>
## Retrieval Enhanced Model for Commonsense Generation

### Results

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU-4</th>
<th>CIDEr</th>
<th>SPICE</th>
<th>SPICE(v1.0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT-Gen (Bao et al., 2020)</td>
<td>23.468</td>
<td>12.606</td>
<td>24.822</td>
<td>27.30</td>
</tr>
<tr>
<td>UniLM (Dong et al., 2019)</td>
<td>30.616</td>
<td>14.889</td>
<td>27.429</td>
<td>30.20</td>
</tr>
<tr>
<td>BART (Lewis et al., 2020)</td>
<td>31.827</td>
<td>13.976</td>
<td>27.995</td>
<td>30.60</td>
</tr>
<tr>
<td>T5-base (Raffel et al., 2020)</td>
<td>18.546</td>
<td>9.399</td>
<td>19.871</td>
<td>22.00</td>
</tr>
<tr>
<td>T5-large (Raffel et al., 2020)</td>
<td>31.962</td>
<td>15.128</td>
<td>28.855</td>
<td>31.60</td>
</tr>
<tr>
<td>EKI-BART (Fan et al., 2020)</td>
<td>35.945</td>
<td>16.999</td>
<td>29.583</td>
<td>32.40</td>
</tr>
<tr>
<td>KG-BART (Liu et al., 2021)</td>
<td>33.867</td>
<td>16.927</td>
<td>29.634</td>
<td>32.70</td>
</tr>
<tr>
<td>CALM(T5-base) (Zhou et al., 2021)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>33.00</td>
</tr>
<tr>
<td>RE-T5 (ours)</td>
<td>40.863</td>
<td>17.663</td>
<td>31.079</td>
<td>34.30</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>SPICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieve (only)</td>
<td>29.60</td>
</tr>
<tr>
<td>T5</td>
<td>30.80^3</td>
</tr>
<tr>
<td>T5 + MR</td>
<td>33.60</td>
</tr>
<tr>
<td>T5 + MR + pretrain</td>
<td>33.90</td>
</tr>
<tr>
<td>RE-T5 (T5 + TR + pretrain)</td>
<td><strong>34.30</strong></td>
</tr>
</tbody>
</table>
Retrieval-based augmentation

- KFCNet:
  
  Li, Haonan, Yeyun Gong, Jian Jiao, Ruofei Zhang, Timothy Baldwin, and Nan Duan. "KFCNet: Knowledge Filtering and Contrastive Learning Network for Generative Commonsense Reasoning." *EMNLP 2021*

- Differentiable open-ended commonsense reasoning

Today: Language models + commonsense reasoning

Outline

- Commonsense reasoning  ✔
- Pre-trained language models  ✔
- The four ways of using PLTM for commonsense reasoning:
  1. Pre-training  ✔
  2. Retrieval-based augmentation  ✔
  3. Model-based augmentation
  4. Formal logic and symbolic reasoning
Model-based augmentation
Model-based augmentation

Overview

- Conceptnet and knowledge bases contain information about a fixed set of entities

- How do we generate information to augment when open-domain events are involved?
  - What happens when you smash a glass on a wooden floor?
  - If someone is wearing sunglasses, is it more likely that rain is falling?

- Language models as knowledge bases

- Our recent works
Explainable defeasible reasoning over graphs using Mixture-of-experts

Aman Madaan, Yiming Yang

Joint work with Niket Tandon, Dheeraj Rajagopal, Peter Clark, Eduard Hovy

https://github.com/madaan/thinkaboutit
Defeasible Reasoning
Defeasible Reasoning

• A classification task

• Given a premise $P$, a hypothesis $H$
  
  • New evidence (update) $U$ may be weaken or strengthen the hypothesis


Defeasible reasoning requires implicit background knowledge.

**Background Knowledge:**
- Floor is carpeted
- Glass dropped in the dining hall
- Floor absorbs shock
- Floor is hard
- Glass dropped on the dance floor
- Floor does not absorb shock
- Glass shatters

**Probabilities:**
- P: Glass falls on the floor
- H: Glass shatters
- U: floor is carpeted

**Analysis:**
- Glass is less likely to break on a carpeted floor.
- Hypothesis: Glass shatters.

**Conclusion:**
- Update: weakens Hypothesis.
Dataset

- A large dataset of defeasible reasoning queries and such graphs are available
- Dataset of defeasible queries [1]
  - Manually created, spans three domains:
    - ATOMIC (Commonsense, 43K)
    - SOCIAL-CHEM-101 (Social norms, 95K)
    - SNLI (NLI, 92K)
  - A lot of implicit knowledge is used for answering these queries
    - Dataset of graphs generated using transfer learning [2]
    - For each defeasible query, the graph captures additional context that can be useful


**How to best use graphs for defeasible reasoning?**

Given a defeasible query $PHU$

And a graph generated for the query (augmented information)

Generate classification label

- Floor is carpeted
- Floor absorbs shock
- Glass shatters

- Glass dropped in the dining hall
- Floor is hard
- Floor does not absorb
- Glass shatters

- Glass dropped on the dance floor
- Update **weakens** Hypothesis

Floor is carpeted **strengthens** Hypothesis

- Glass dropped in the dining hall
- Floor is hard
- Floor does not absorb
- Glass shatters

- Glass dropped on the dance floor
- Update **weakens** Hypothesis

48
Baselines

• **W/O G:** Concatenate *PHU* as a single string and fine-tune (Rudinger et al. 2020)

• **STR:** Append G in a string format after *PHU*, and fine-tune
  • Break down the graph into node - edge - node triplets
  • Append the triplets with the query as a string

• Use RoBERTa as the encoder
Results

Baselines

<table>
<thead>
<tr>
<th></th>
<th>ATOMIC</th>
<th>SNLI</th>
<th>SOCIAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>W/O G</td>
<td>75</td>
<td>75</td>
<td>86.25</td>
</tr>
<tr>
<td>STR</td>
<td>80</td>
<td>82.5</td>
<td>86.25</td>
</tr>
<tr>
<td>STR</td>
<td>85</td>
<td>78.75</td>
<td>82.5</td>
</tr>
<tr>
<td></td>
<td>90</td>
<td>90</td>
<td>75</td>
</tr>
</tbody>
</table>
Performance: can we do better?

Explainability: can we identify which parts of the graph are more useful?
How to best use graphs to increase performance on the task?

- **STR**: Discards the semantics of various parts of $G$ (mediator, external situation etc.)

- From human evaluation:
  - **Not every part** of the augmented graph was useful
  - The augmented graph was **not always useful**

- Thus the model needs to be able to:
  - **Selectively use** parts of the input
  - **Discard** augmentation completely
Mixture of experts for pooling graph representations

Results

Atomic

<table>
<thead>
<tr>
<th>Method</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>W/O G</td>
<td>78.75</td>
</tr>
<tr>
<td>STR</td>
<td>79.5</td>
</tr>
<tr>
<td>GCN</td>
<td>80.25</td>
</tr>
<tr>
<td>MoE</td>
<td>81</td>
</tr>
</tbody>
</table>

SNLI

<table>
<thead>
<tr>
<th>Method</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>W/O G</td>
<td>81</td>
</tr>
<tr>
<td>STR</td>
<td>82.25</td>
</tr>
<tr>
<td>GCN</td>
<td>83.5</td>
</tr>
<tr>
<td>MoE</td>
<td>84.75</td>
</tr>
</tbody>
</table>

SOCIAL

<table>
<thead>
<tr>
<th>Method</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>W/O G</td>
<td>86</td>
</tr>
<tr>
<td>STR</td>
<td>86.75</td>
</tr>
<tr>
<td>GCN</td>
<td>88.25</td>
</tr>
<tr>
<td>MoE</td>
<td>89</td>
</tr>
</tbody>
</table>
Situation nodes are more important

Select relevant graph nodes

MOE-V

h_G

C- C+ U S- M- M+ H- H+

ATOMIC

SNLI

SOCIAL

Contextualizers

Situation

Mediators
Graphs are more useful for strengthens questions

Select whether to use graph vs. question

MOE-GX

$h_G$ $h_X$

$p_G$ $p_X$

Strenghtens

Weakens
MOE-V learns the node semantics

More discussion on explainability in the paper!
Summary

- Thinking about a question scenario before modeling it helps the models.
- Mixture-of-experts allows effective and explainable learning over graphs.
- For KAIROS, similar strategies can be used to highlight the part of schemas that were used in matching or prediction.

Code, pre-trained models, data for the EMNLP 2021 paper: https://github.com/madaan/thinkaboutit
Today: Language models + commonsense reasoning

Outline

- Commonsense reasoning ✔
- Pre-trained language models ✔
- The four ways of using PLTM for commonsense reasoning:
  1. Pre-training ✔
  2. Retrieval-based augmentation ✔
  3. Model-based augmentation ✔
  4. Formal logic and symbolic reasoning
Formal logic and symbolic reasoning
BeliefBank: Adding Memory to a Pre-Trained Language Model for a Systematic Notion of Belief

Nora Kassner\textsuperscript{1,2}, Oyvind Tafjord\textsuperscript{1}, Hinrich Schütze\textsuperscript{2}, Peter Clark\textsuperscript{1}

\textsuperscript{1}Allen Institute for AI, Seattle, WA
\textsuperscript{2}Center for Information and Language Processing, LMU Munich, Germany

kassner@cis.lmu.de
{oyvindt,peterc}@allenai.org

EMNLP 2021
BeliefBank: Adding Memory to a Pre-Trained Language Model for a Systematic Notion of Belief

Overview

- Language models are getting bigger to the point that even fine-tuning is intractable

- Can we add formal constraints on the model to improve its performance?

- Test of a consistent belief (e.g., “eagles are birds”)
  - Re-phrasings are Are eagles birds? Is an eagle a type of bird?
  - Consistently talk about all the downstream tasks
BeliefBank: Adding Memory to a Pre-Trained Language Model for a Systematic Notion of Belief

Definitions

- **Belief**: a weighted triple \((s, l, w)\)
  - \(s\) is a sentence (*a poodle is a dog*)
  - \(l\) is the label \(\in\) \{true, false\} (*true*)
  - \(w\) is system’s strength of the belief (*0.9*)

- **Belief-bank**: a set of beliefs

- **Constraint**: a 5 tuple \((s_i \cdot l_i \rightarrow s_j \cdot l_j, w_i)\)
  - Connects two beliefs with a weight if they are violated.
  - “X is a dog”.T \(\rightarrow\) “X has a tail”.T, 0.8
    - Dogs usually have a tail
  - “X is a bird”.T \(\rightarrow\) “X is a fish”.T, 1.0
    - A fish cannot be a bird

- **Consistency**:
  - \(\tau = |\{ c_i \mid \neg(s_i \cdot l_i \rightarrow s_j \cdot l_j) \}| / |\{ c_i \mid s_i \cdot l_i \}|\)

<table>
<thead>
<tr>
<th>Beliefs</th>
<th>12.5k</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constraints</td>
<td>2600</td>
</tr>
</tbody>
</table>
BeliefBank: Adding Memory to a Pre-Trained Language Model for a Systematic Notion of Belief

Overview

- Given:
  - A *stream* of sentences $Q$, each with a truth value (i.e., true or false)
  - A set of constraints $C(s)$ between sentences in $Q$, each with a penalty $w$
  - A *Model* $M$ that maps $Q \rightarrow \{true, false\}$
  - A *SAT solver*, that can flip the truth value of sentences to incur minimum penalty

- Task:
  - Accumulate the labels for $Q$ as predicted by $M$, so that they are globally consistent
A swallow is a fish? 

Model: 

- A swallow is a bird: T
- A swallow has fur: F
- A swallow has wings: T

Belief bank: 

- A swallow is a fish: F
- A swallow has fur: F
- A swallow has wings: T

SAT Solver: 

Constraints: 

Just asking Do swallows have gills? Leads to True!
Method

(A) raw model

(B) constraint-solving

(C) feedback

(D) feedback + constraint-solving
Results

**OnTopic-FB** = using (randomly selected) on-topic feedback from old answers for new queries.

**Relevant-FB** = using most relevant on-topic feedback for new queries.

**Constraints** = running the constraint-solver after each batch.

Figure 3: Accuracy (left) and consistency (right) of the growing BeliefBank, as the system answers incrementally more questions (each batch = 10% of the queries). Relevant feedback, constraint-solving, and both, all help improve both F1 and Consistency.
Formal logic and symbolic reasoning

Additional references

- Conversational Multi-Hop Reasoning with Neural Commonsense Knowledge and Symbolic Logic Rules

  Tom Mitchell


- Improving GPT-3 after deployment with a dynamic memory of feedback

  https://openreview.net/forum?id=6DBkg64mzt6
Today: Language models + commonsense reasoning

Outline

- Commonsense reasoning  
- Pre-trained language models  
- The four ways of using PLTM for commonsense reasoning:
  1. Pre-training  
  2. Retrieval-based augmentation  
  3. Model-based augmentation  
  4. Formal logic and symbolic reasoning
Do the models really have commonsense?
Clever Hans

- Giving right answer for the wrong reasons?

- Are the models *really* doing commonsense reasoning?

- Does it even matter?
Back to Square One:
Artifact Detection, Training and Commonsense Disentanglement
in the Winograd Schema

Yanai Elazar$^{1,2}$ Hongming Zhang$^{3,4}$ Yoav Goldberg$^{1,2}$ Dan Roth$^4$
$^1$Bar Ilan University, $^2$AI2, $^3$HKUST, $^4$UPenn
{yanaiela,yoav.goldberg}@gmail.com
hzhangal@cse.ust.hk, danroth@seas.upenn.edu

EMNLP 2021
Back to Square One: Artifact Detection, Training and Commonsense Disentanglement in the Winograd Schema

Overview

*The trophy doesn’t fit into the brown suitcase because it is too large.*

*The trophy doesn’t fit into the brown suitcase because it is too small.*
Back to Square One: Artifact Detection, Training and Commonsense Disentanglement in the Winograd Schema

Overview

- Test if the model is giving the right answer for the right reasons

- If the model really knew
  
  • It should have no trouble getting both the questions in the pair right
  
  • Performance on questions that do not have enough information should be random
# Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Setup</th>
<th>Single</th>
<th>Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSC</td>
<td>original</td>
<td>89.71</td>
<td>79.41</td>
</tr>
<tr>
<td></td>
<td>no-cands</td>
<td>60.72</td>
<td>40.35</td>
</tr>
<tr>
<td></td>
<td>part-sent</td>
<td>64.88</td>
<td>33.88</td>
</tr>
<tr>
<td>WSC-na</td>
<td>original</td>
<td>89.45</td>
<td>79.09</td>
</tr>
<tr>
<td></td>
<td>no-cands</td>
<td>58.06</td>
<td>34.41</td>
</tr>
<tr>
<td></td>
<td>part-sent</td>
<td>59.90</td>
<td>25.00</td>
</tr>
<tr>
<td>Winogrande</td>
<td>original</td>
<td>71.49</td>
<td>58.45</td>
</tr>
<tr>
<td></td>
<td>no-cands</td>
<td>53.07</td>
<td>31.05</td>
</tr>
<tr>
<td></td>
<td>part-sent</td>
<td>53.11</td>
<td>22.34</td>
</tr>
</tbody>
</table>
Do the models really have commonsense? models really have commonsense?

Additional references


What’s next?

- Exploring what exactly are these large language models learning?
- How much data do they need to generalize?
- How does that knowledge transfer to the real world?
- Interactive learning
- Multi-modal commonsense reasoning
Language models + commonsense reasoning

Summary
- Using large pre-trained language models (PTLM) for commonsense reasoning
- The four paths to commonsense reasoning:
  1. Pre-training
     Pre-train with novel objectives
  2. Retrieval-based augmentation
     Supplement LM with additional information
  3. Model-based augmentation
     Use another model to generate open-ended augmentation
  4. Formal logic and symbolic reasoning
     Drastically different techniques, not everything is an embedding
- Do the models really have commonsense?
  - Depends on the definition
  - Probably not (yet), but more investigation is needed