自然语言处理期末作业

基于大语言模型的思维链推理

第十小组成员：黄倪远；谢绎含；单歆芮；陈逸晗；庞瑛霏

指导老师：汤斯亮

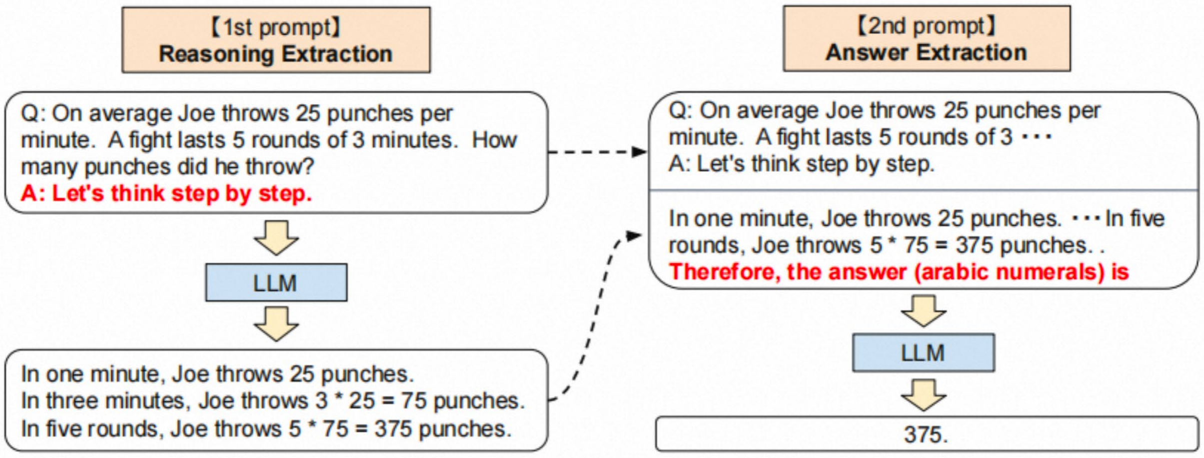
目 录

CONTENTS

第一单元
CoT技术及其挑战

CoT技术

　　思维链推理（Chain of Thought，CoT）是一个与人工智能领域的自然语言处理技术相关的概念，尤其是在解决复杂问题或执行任务的上下文中。CoT通常指的是一系列逻辑步骤，表现为在达到最终答案或解决方案之前，明确地阐述问题解决过程中的每一步骤。



CoT技术的优势

CoT作为人工智能利于的一个研究方向和实践方法已经展现出了潜力，尤其在处理复杂任务和问题解答的环境中，具体体现在：

1. **增强问题解决能力**：CoT在多步骤推理和解决需要连锁推理的复杂问题中表现出色，如数学问题、物理问题、编程挑战等。

2. **提高解释性**：通过阐述解决问题的每个步骤，CoT增强了AI的可解释性和透明度，使得用户理解AI决策过程变得更加容易。

3. **错误检测和修正**：CoT提供的步骤可以帮助识别和修正推理过程中出现的逻辑错误或计算错误。

CoT技术的优势

尽管CoT能够提供透明的推理链条，但这并不能保证推理的结论总是正确的。AI仍有可能在某些步骤中产生错误，进而引导到错误的结论。阻碍CoT准确性的因素可能包括但不限于：

1. **不完整或错误的知识库：** AI模型依赖于它们训练时接触的数据。如果这些数据中含有不正确或不完整的信息，它可能在推理过程中得出错误的结论。

2. **长推理链的问题累积：** 在处理长推理链时，每一步的小错误都可能累积起来导致最终结论的显著偏差。

3. **逻辑错误：** 在构建推理链的过程中，AI可能会犯逻辑上的错误，比如错误地应用假设、规则或定律，或者对因果关系做出错误的推断。

第贰单元

Multi LLM consistency

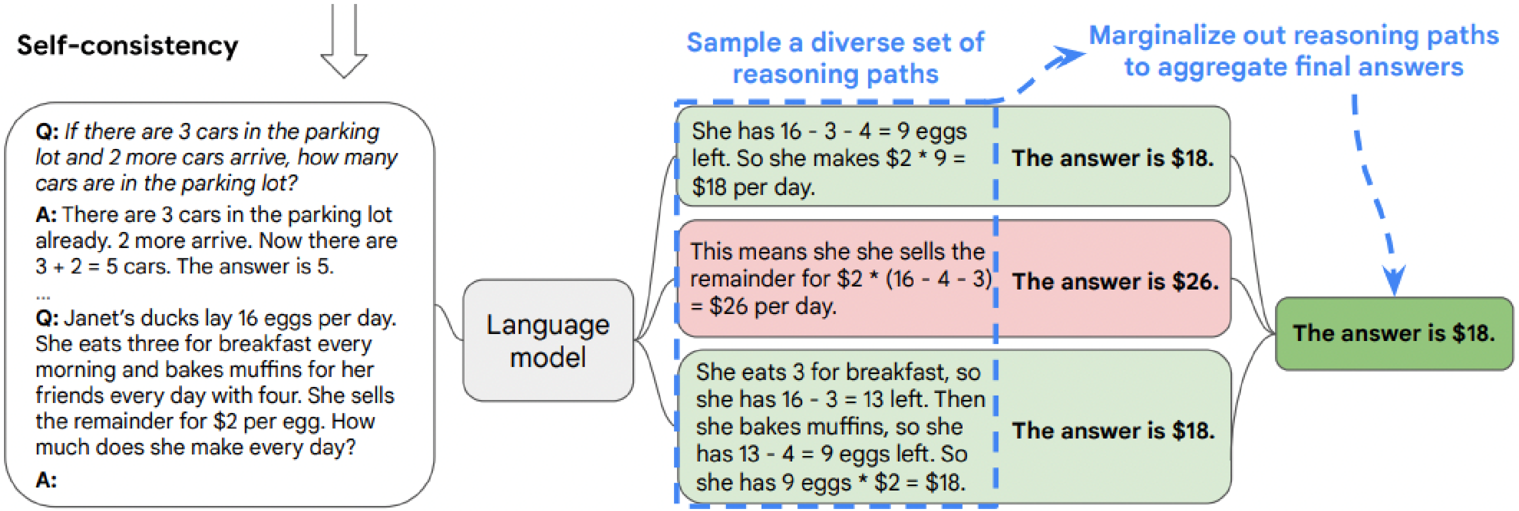
算法介绍

Multi LLM consistency算法通过整合跨多个大型语言模型（LLMs）的推理路径来提高输出结果的一致性和准确性。该算法的主要步骤如下：

1. 在多个不同的大型语言模型（如GPT4、PaLM2等）上运行相同的思考链提示，以采样出不同的推理路径。这一步骤旨在从多样化的角度探索问题，以便捕捉到可能的不同解决方案和思维方式。

2. 确立共识（quorum），选择最佳响应。在此示例中，使用GPT4作为共识评估器，但也可以采用其他方法。共识的建立是通过比对不同模型给出的答案，从中选出最为一致或出现频率最高的答案，以此作为最终的输出。

算法介绍



Self-Consistency Improves Chain of Thought Reasoning in Language Models

实验步骤及结果

1. input为： Q: When I was 6 my sister was half my age. Now I'm 70 how old is my sister? A:
2. 将input分别输入GPT-4, CHATGPT, PALM三个模型中，输出为：
   - GPT-4: When you were 6 your sister was 3. Therefore, she is 3 years younger than you. So if you are 70 now, your sister is 70 - 3 = 67 years old.
   - CHATGPT： If when you were 6, your sister was half your age, that means when you were 6, your sister was 3 years old. The age difference between you and your sister is always the same, so when you're 70, your sister would be 70 - 3 = 67 years old.
   - PLAM: When you were 6 your sister was 6 / 2 = 3 years old. So your sister is 70 - 3 = 67 years old. The answer is 67.
3. 最终结果为： The majority across the outputs indicates that the sister is 67 years old.

实验步骤及结果

```python
# Run the AI Config
from aiconfig.default_parsers.parameterized_model_parser import InferenceOptions

params = {}
inference_options = InferenceOptions()

response_completion = await config.run("majority-evaluator", params, inference_options, run_with_dependencies=True)
response = config.get_output_text("majority-evaluator")
```

```
When you were 6, your sister was half your age, which means she was 3 years old. So she's 3 years younger than you. If you're
```

```python
print(response)
```

```
The majority across these outputs is the first one, with a calculation stating that the sister is 67 years old.
```

实验结果分析

该实验是选择使用不同的LLM去生成多条推理链，该方法可以使得答案错误的概率较之只选用单一模型降低。

然后该方法有着几个明显的弊端：

1. 推理链的生成受限于大语言模型的数量；
2. 多数即为答案的并不总是有效，有时候不同的语言模型会生成各不相同的答案，将没办法选出正确的答案是什么。

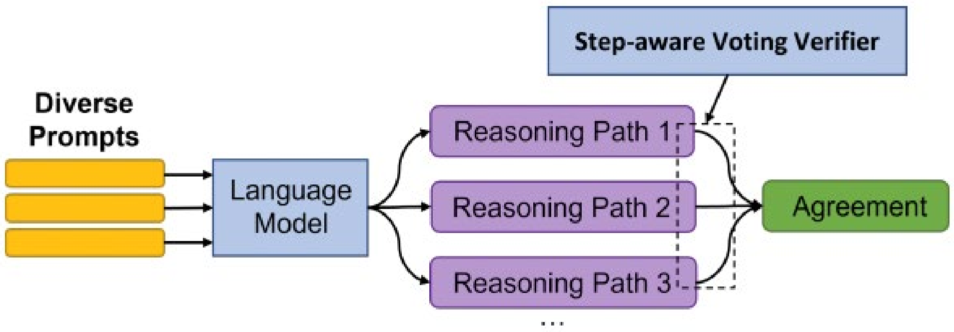
第叁单元

DIVERSE

算法介绍

DIVERSE（Diverse Verifier on Reasoning Step）是一种新颖的方法，旨在增强语言模型的推理能力。该算法主要包括三个组成部分：生成多样化提示以探索不同的推理路径、**使用验证器根据加权投票方案过滤不正确的答案**、以及逐个验证每个推理步骤而不是整个链条。通过这些组件，DIVERSE能够在推理任务中引导语言模型，提高其性能并取得新的最先进结果。

算法介绍



Making Large Language Models Better Reasoners with Step-Aware Verifier

实验步骤及结果

- 数据集：GSM8K
- Input question: Pauline has 125 matchbox cars. They are all either convertibles, trucks, regular cars. 64% of them are regular cars. 8% are trucks. How many convertibles does she own?
- Truth: The trucks and regular cars make up 72% of her collection because 64+8 equals <<64+8>>72%%This means convertible make up 28% of her collection because 100-72 equals 28.%%She has 35 convertibles because 125 times .28 equals <<125*.28=35>>35%%### 35
- sample0: There are 125 cars in total%%64% of them are regular cars so 64/100*125= <<64/100*125=80>>80 of them are regular cars%%8% of them are trucks so 8/100*125= <<125-80-10=35>>35 of them are convertibles%%### 35

评价指标及结果

random_top1:是通过对数据集进行随机采样，然后计算随机选择的预测结果与真实答案匹配的频率来计算的。

voting_top1_accuracy:基于多数投票来评估预测性能的。对于每个样本，它计算预测结果中哪个答案被选为最多的次数，并将其与真实答案匹配。

recall:是召回率，用于评估模型能够正确预测出真实答案的能力。

	训练集	测试集
random_top1	0.565	0.576
voting_top1_accuracy	0.783	0.782
recall	0.941	0.955

实验结果

```
(microsoft.msra.dki.verifier_data_preparing) [ma-user src]$python verifier_data_prepare.py --generator_result_file executive/tra
in_dir --output_dir executive/train_preprocessed --split train --random_seed 233 --dataset_name GSM8K
/home/ma-user/anaconda3/envs/PyTorch-1.10.2/lib/python3.7/site-packages/requests/__init__.py:104: RequestsDependencyWarning: url
lib3 (1.26.12) or chardet (5.2.0)/charset_normalizer (2.0.12) doesn't match a supported version!
  RequestsDependencyWarning)
100%|████████████████████████████████████████| 20000/20000 [00:00<00:00, 200268.53it/s]
Total cases: 1000
Case 0's question: Pauline has 125 matchbox cars. They are all either convertibles, trucks, regular cars. 64% of them are regula
r cars. 8% are trucks. How many convertibles does she own?
Case 0's ground truth: The trucks and regular cars make up 72% of her collection because 64+8 equals <<64+8=72>>72%%This means c
onvertibles make up 28% of her collection because 100-72 equals 28.%%She has 35 convertibles because 125 times .28 equals <<125*
.28=35>>35%%#### 35
Case 0's sample0: There are 125 cars in total%%64% of them are regular cars so 64/100*125=<<64/100*125=80>>80 of them are regula
r cars%%8% of them are trucks so 8/100*125=<<8/100*125=10>>10 of them are trucks%%The remaining cars are convertibles so 125-80-
10=<<125-80-10=35>>35 of them are convertibles%%#### 35
*********** Data statistics ***********
random_top1: 0.565
voting_top1_accuracy: 0.783
recall: 0.941

100%|████████████████████████████████████████| 1000/1000 [00:01<00:00, 592.95it/s]
100%|████████████████████████████████████████| 1000/1000 [00:01<00:00, 976.66it/s]
100%|████████████████████████████████████████| 1000/1000 [00:00<00:00, 192258.16it/s]
100%|████████████████████████████████████████| 21000/21000 [00:00<00:00, 35624.84it/s]
```

实验结果

```
(microsoft.msra.dki.verifier_data_preparing) root@autodl-container-bb8c4f8e12-349f008e:~# python DIVERSE/code/src/verifier_data_prepare.py --generator_result_file exec_dir/test_dir --output_dir exec_dir/test_preprocessed --split dev --random_seed 233 --dataset_name GSM8K
100%|████████████████████████████████████████| 99740/99740 [00:00<00:00, 233324.51it/s]
Total cases: 998
Case 0's question: Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day at the farmers' market?
Case 0's ground truth: Janet sells 16 - 3 - 4 = <<16-3-4=9>>9 duck eggs a day.%%She makes 9 * 2 = $<<9*2=18>>18 every day at the farmer's market.%%#### 18
Case 0's sample0: She eats 3 and uses 4, so that is 7 eggs.%%16-7=<<16-7=9>>9 eggs to sell.%%9 eggs*$2=<<9*2=18>>$18/day%%#### 18
*********** Data statistics ***********
random_top1: 0.5761523046092184
voting_top1_accuracy: 0.781563126252505
recall: 0.9549098196392786

100%|████████████████████████████████████████| 998/998 [00:05<00:00, 187.83it/s]
100%|████████████████████████████████████████| 998/998 [00:01<00:00, 505.74it/s]
100%|████████████████████████████████████████| 998/998 [00:00<00:00, 88980.62it/s]
100%|████████████████████████████████████████| 40918/40918 [00:01<00:00, 34566.20it/s]
(microsoft.msra.dki.verifier_data_preparing) root@autodl-container-bb8c4f8e12-349f008e:~#
```

实验结果分析

该方法效果很好，在多数投票来评估已经能显著提高准确率的情况下，再一次显著将准确率提高。在训练集和测试集均有着不错的表现。

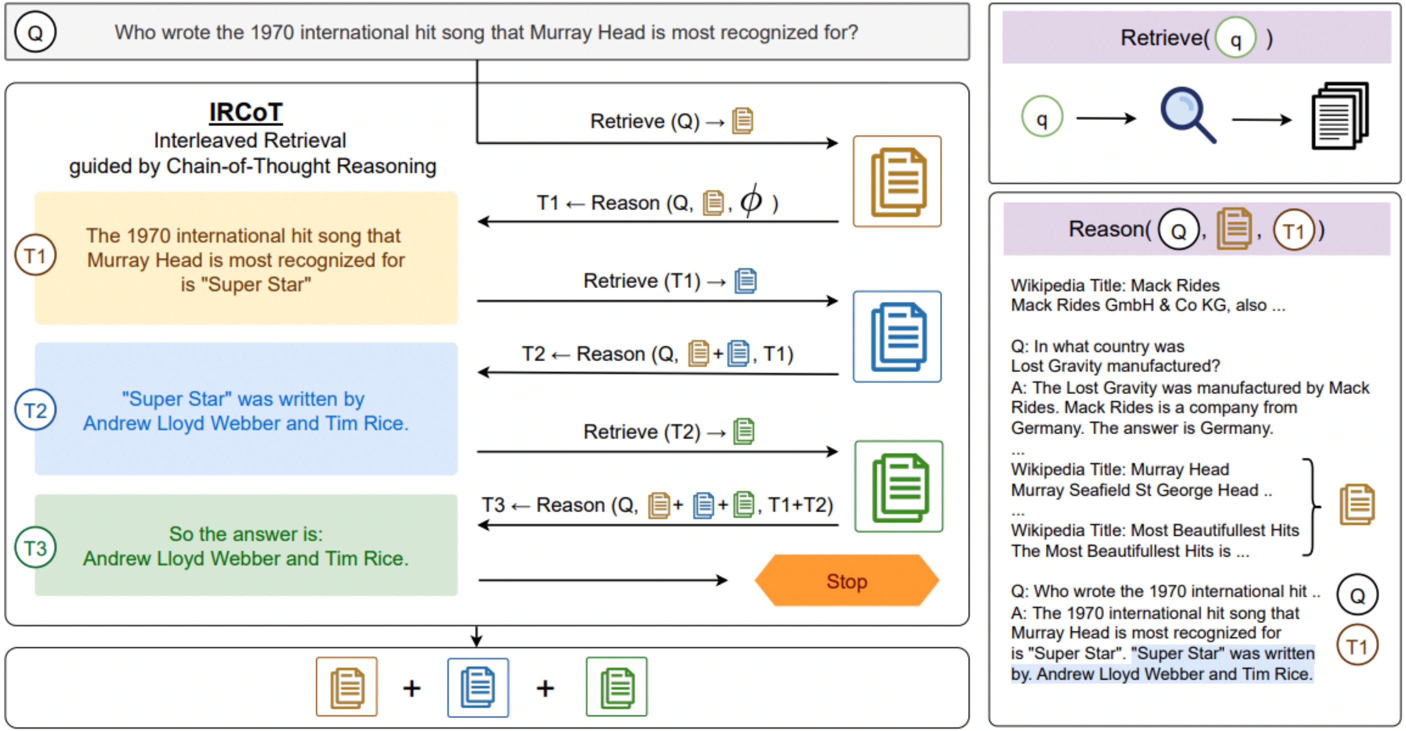
第肆单元
IRCoT

算法介绍

IRCoT，即 Interleaving Retrieval with Chain-of-Thought Reasoning，是一个基于结合检索（Retrieval）与链式推理（Chain-of-Thought Reasoning）的框架，它专为解决知识密集型的多步问题设计。这种方法通过在检索外部知识与执行链式推理之间交错执行，旨在提高人工智能系统解决复杂问题的能力。

在传统的链式推理过程中，模型通常依赖于内部知识库或之前训练中获取的知识来解答问题。然而，当面临知识密集型的问题时，内部知识库可能不够全面或更新不及时，从而限制了推理的准确性和可靠性。

算法介绍



Interleaving Retrieval with Chain-of-Thought Reasoning for Knowledge-Intensive Multi-Step Questions

实验步骤及结果

- 实验步骤大致如下（详细步骤可参考./code/3_IRCoT/README.md)
  1. 配置实验环境：python 3.8.0
  2. 准备数据集： hotpotqa，2wikimultihopqa，musique，iirc
  3. 准备prompt
  4. 准备Retriever和LLM Servers
     - SYSTEM: choose from (ircot，ircot_qa，oner，oner_qa，nor_qa)
     - MODEL: choose from (codex，flan-t5-xxl，flan-t5-xl，flan-t5-large，flan-t5-base，none)
  5. 输入问题
  6. 提取出问题中较为陌生的词语去检索
  7. 将检索出的词语带进问题，对问题进行推理
  8. 重复步骤6，7直到得出答案

实验步骤及结果

What is the length of the river into which Pack Creek runs after it goes through the Spanish Valley?

A: ["Spanish Valley", "Spanish Creek (Plumas County, California)"]

A: Pack Creek runs through the Spanish Valley.

A: Exit? No.

A: ["Spanish Valley", "Spanish Creek (Plumas County, California)", "Polk Valley Run"]

实验步骤及结果

A: ["Spanish Valley", "Spanish Creek (Plumas County, California)", "Polk Valley Run", "Missouri River Valley"]

A: Pack Creek runs through the Spanish Valley. After it goes through the Spanish Valley, it flows north northwestward into Moab Valley toward its confluence with the Colorado River. The Colorado River is approximately 1,450 miles long.

A: Exit? No.

A: ["Spanish Valley", "Spanish Creek (Plumas County, California)", "Polk Valley Run", "Missouri River Valley", "Blue River (Colorado)", "Eagle River (Colorado)"]

A: Pack Creek runs through the Spanish Valley. After it goes through the Spanish Valley, it flows north northwestward into Moab Valley toward its confluence with the Colorado River. The Colorado River is approximately 1,450 miles long. So the answer is: 1,450 miles.

实验结果分析

对于一些题目中蕴含复杂名词的问题，该方法可以借助检索来提高答案的生成。

但是该方法的成功率不高，有相当一部分问题没办法检索出问题当中陌生名词的信息从而无法进行推理。该方法同样强烈依赖检索器的质量。

实验步骤及结果

```
****************************************
Experiment Name: ircot_qa_codex_hotpotqa
****************************************
python run.py summarize ircot_qa_codex_hotpotqa --instantiation_scheme ircot_qa --
prompt_set 1 --evaluation_path processed_data/hotpotqa/dev_subsampled.jsonl

bm25_retrieval_count  distractor_count       metric_value
0                2          "1"  60.5 | 64.6 | 61.4 |   100
1                2          "2"  60.9 | 64.0 | 62.8 |   100
2                2          "3"  60.1 | 63.7 | 61.2 |   100
3                4          "1"  55.6 | 58.8 | 57.3 |   100
4                4          "2"  57.6 | 61.1 | 58.7 |   100
5                4          "3"  55.3 | 58.8 | 56.7 |   100
6                6          "1"  55.6 | 59.1 | 56.8 |   100
7                6          "2"  52.2 | 55.6 | 53.5 |   100
8                6          "3"  48.9 | 52.0 | 50.2 |   100
9                8          "1"  53.5 | 56.4 | 54.7 |   100
10               8          "2"  52.7 | 55.6 | 53.6 |   100
...
```

感谢您的观看

第十小组成员：黄倪远；谢绎含；单歆芮；陈逸晗；庞瑛霏

指导老师：汤斯亮