

0.1. Автор *Маканов А.Ж.* Способы повышения эффективности Chain of Thoughts prompting

1 Введение

В данном небольшом исследовании сравнивается стандартная реализация CoT [1] с его ансамблированным вариантом [2] на датасете GSM8K. Тестировались различные техники генерации текста с помощью распределенной модели BLOOM. Также был предложен метод по улучшению ансамблирования метода CoT.

В качестве подсказок (prompts) были выбраны 8 подсказок, приведенными в статье [1], где они использовались для датасетов с текстовым математическими задачами. Они все представлены в Таблице 4.

2 Постановка эксперимента

Были проведены эксперименты на моделях Bloom 7b и Bloom 176B, при различных режимах генерации, объемах выборки и метода CoT.

1. BLOOM 176B. Протестирован стандартный режим CoT (на ансамблированный вариант не осталось времени) Размер выборки - 15 примеров. Было опробовано 2 режиме генерации - sampling и top-k.
2. BLOOM 7B. Протестирован стандартный, ансамблированный режимы CoT. Для стандартного CoT размер выборки - 100, в других случаях - 30. Режимы генерации - sampling, top-k, top-p.

Каждый эксперимент повторялся дважды с различным seed, чтобы результаты были надежными.

3 Методы генерации текста

В процессе работы исследовались последствия применения различных методов генерации:

1. sampling
2. top-k (k=50)
3. top-p (p=0.9)

Temperature, beam search не рассматривались, из-за ограниченности ресурсов и времени. На ранних экспериментах greedy метод генерации показал свою несостоятельность - довольно часто он закликивался и наблюдалось такое явление, как inductive bias. Это сильно тормозило процесс. Было принято решение его не рассматривать.

Результаты работы представлены в Таблицах 1 и 2 Для интерпретируемости результатов был внесен стохастический элемент: каждый эксперимент повторялся дважды с различным seed. Результаты, представленные в Таблицах, являются усредненными.

Как указывается в статье [2], метод ансамблирования Unweighted sum (majority vote), на удивление является самым очевидным и самым эффективным вариантом. Во время ансамблирования из модели извлекалось только 5 ответа, также, из-за ограниченности ресурсов. В той же самой статье приводится, что заметное улучшение качества предсказания наблюдается только при количестве Reasoning path > 4,

4 Разработанный метод модификации

В качестве метода улучшения ансамблированного CoT я предлагаю подбирать такие подсказки (prompts), которые бы были семантически близки в примеру. Т.о. модель усваивает контекст тематики задач, и гипотетически способна решать их лучше, получив доступ к нескольким решениям схожих задач. Семантическая близость измеряется с помощью эмбедингов Doc2Vec. Контекстные векторы документов (математических задач) можно получить с помощью BERT-base [3]. Как описывается в статье [4], для получения лучших контекстных эмбедингов следует рассматривать скрытые состояния, находящиеся ближе к середине, чем к концу модели (был взят 7-ой слой). После получения контекстных векторов всех доступных подсказок (их 8) и контекстного вектора примера, измеряется косинусное подобие между этими документами. Пять самых похожих подсказки используются для ансамблированного CoT.

5 Получение ответа

В процессе изучения работы модели BLOOM было замечено, что представляется затруднительным доставать из CoT окончательный ответ. Далеко не всегда размышления модели оканчивались желанным "answer is X". В данных такое встречалось только в 14% случаев. По этой причине было принята попытка расширить процесс генерации ответа вторичной обработкой последовательности: < PROMPT+Q+A+The answer is: >

Задача этой обработки - обобщить Chain of Thoughts модели, и вычленив из нее итоговый ответ. В Таблице 1 приведены результаты оценки, полученной с помощью вторичной обработки, и в которых итоговый ответ получается из сгенерированной последовательности с помощью парсинга (берем последнее встречающееся целое число), или же число, попадающее в шаблон "answer is".

Вторичная обработка имеет свои ограничения, поскольку требует еще раз скормить последовательность токенов модели, причем эта последовательность ощутимо больше, что в может привести

к превышению лимита токенов, подаваемых модели и к аварийной остановке генерации.

Генерация	CoT	ansCoT
sample-prs	2.0±2.0	0.0±0.0
sample-sec	2.0±1.0	3.3±3.3
top-k-prs	2.0±0.0	3.9±2.7
top-k-sec	1.5±1.5	0.0±0.0
top-p-prs	2.5±0.5	3.3±3.3
top-p-sec	0.5±0.5	0.0±0.0

Таблица 1: Bloom 7B. Точности предсказания стандартного CoT, ансамблированного CoT и модифицированного CoT. Ответ вычленился с помощью вторичной обработки - sec и помощью парсинга - prs.

Генерация	CoT
sample-prs	7.1±0.5
sample-sec	3.3±3.3
top-k-prs	6.2±6.2
top-k-sec	6.2±6.2

Таблица 2: Bloom 176B. Точности предсказания стандартного CoT.

6 Результаты

В Таблице 1 представлены результаты экспериментов над способами генерации и реализации Chain of Thought Prompting с моделью Bloom 7b. В Таблице 2 - с моделью Bloom 176B. К сожалению, результаты ансамблированного варианта для большой модели посчитать не хватило времени. Для стандартного метода CoT лучше всех справился режим генерации - top-p. Для ансамблированного CoT лучше всего работал top-k. Видно, что модель с большим количеством параметров стабильно справляется лучше. В случае большой модели, для стандартной реализации CoT лучшие результаты показала генерация sampling.

Как указывается в статье [2], модель LaMDA-137B, с количеством параметров, сравнимым с BLOOM (176B параметров), достигает точности **17.1%** при стандартном CoT, и **27.7%** при ансамблированном CoT. Результаты, полученные мной, примерно в 4 раза меньше этих результатов. Возмож-

но, причиной этому является то, BLOOM обучался на корпусе из текста web-страниц, в которых диалоги - всего лишь малая часть, в то время как LaMDA изначально проектировалась как модель для поддержания диалога [5] и обучалась на соответствующих данных. Поскольку формат задачи **< PROMPT+Q+A+The answer is: >** очень напоминает диалог, разумно предположить, что LaMDA имеет больше шансов справиться с поставленной задачей.

Также, небольшое отличие ансамблированного и стандартного вариантов CoT можно объяснить тем, что количество извлекаемых ответов для ansamble CoT не такое большое - всего лишь 5. Т.о. если все 5 ответа разные (это наблюдалось в 68% случаев для ансамблированного варианта), то работа модели эквивалентна стандартному режиму.

Вторичная обработка в большинстве случаев не показала своей эффективности, это видно в колонке ansCoT Таблицы 1.

7 Анализ результатов

При ансамблированном CoT правильный ответ содержится в ответах-кандидатах в 11% случаев.

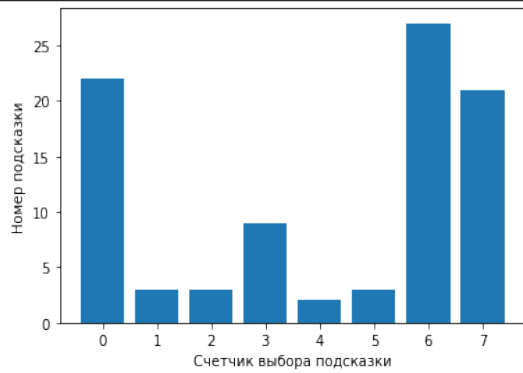
Как указывалось в статье [1], чем длинее CoT, тем вероятнее ответ становился правильным. Результаты экспериментов представлены в Таблице 3. Чем длинее ответ, тем более вероятнее, что он приведет к правильному итоговому выводу. Длинные CoT формализуют собой содержательные, более осмысленные цепочки мыслей, которые наиболее вероятно приводят к истинному ответу.

Правильный	Неправильный
41.69	41.15

Таблица 3: Средние длины CoT в словах, соответствующие правильным и неправильным ответам.

На Рис. 1 показаны частоты, с которыми конкретные подсказки (точнее их часть с условием задачи) выбираются, как семантически более похожие к элементам микродатасета. Видно, что с большим отрывом выделяются первая и последние, а 1, 2, 4, 5 - ые выбирались редко.

Среди решенных задач некоторые встречаются чаще других. Гипотетически, более простые задачи, вероятнее, состоят из меньшего количества слов. Если взять выборку эксперимента для стандартной реализации CoT на Bloom 7B, то задачи три задачи - решены целых 2 раза, и 7 - один раз. Было рассчитано, что средняя длина вопроса у решенных задач - 42.5, в то время как средняя длина (в словах) вопроса в выборке - 48.7.



- LANGUAGE UNDERSTANDING REPRESENTATIONS
<https://arxiv.org/abs/1810.04805>
 [4] LIU ET AL., 2019 LINGUISTIC KNOWLEDGE AND TRANSFERABILITY OF CONTEXTUAL REPRESENTATIONS <https://arxiv.org/pdf/1903.08855>
 [5] THOPPILAN ET AL., 2022 LAMDA: LANGUAGE MODELS FOR DIALOG APPLICATIONS <https://arxiv.org/abs/2201.08239>

Рис. 1: Счетчики частот выбора подсказок, по горизонтали - индексы подсказок, как в Таблице 4.

8 Дальнейшая работа

В качестве дальнейшей работы, разумеется, можно провести все эти эксперименты не на минидатасете, а на большей части набора данных, для получения более надежных результатов. Для модифицированного ансамблированного варианта можно попробовать брать контекстные эмбединги не из 7-ого слоя BERT, а из какого-нибудь другого; использовать другие метрики схожести документов; брать не 5 самых успешных подсказок, а большее количество; выбирать наиболее семантически близкие подсказки случайным образом из распределения, рассчитанного из cosine similarity scores подсказок. Вторичная обработка последовательности может быть модифицирована - вместо того, чтобы сообщать модели `< PROMPT+Q+A+The answer is: >`, Можно подавать:

1. `< Q+A+The answer is: >`
2. `< A+The answer is: >`

Поскольку задача вторичной обработки - суммаризация CoT, то **PROMPT** или **Q** здесь может оказаться абсолютно лишними, и лишь мешать генерализации информации, содержащейся в **A**. Разумеется, можно попробовать брать более масштабные модели. Можно экспериментировать с режимом генерации во время вторичной обработки - greedy метод, например, способен генерировать более точные ответы, если речь заходить о математических задачах, однако склонен к созданию повторяющихся результатов.

Список литературы

- [1] WEI ET AL. 2022 CHAIN-OF-THOUGHT PROMPTING ELICITS REASONING IN LARGE LANGUAGE MODELS <https://arxiv.org/abs/2201.11903>
- [2] WANG ET AL. 2022 SELF-CONSISTENCY IMPROVES CHAIN OF THOUGHT REASONING IN LANGUAGE MODELS <https://arxiv.org/abs/2203.11171>
- [3] DELVIN ET AL., 2019 BERT: PRE-TRAINING OF DEEP BIDIRECTIONAL TRANSFORMERS FOR

Таблица 4: Использованные подсказки

0. **Q:** There are 15 trees in the grove. Grove workers will plant trees in the grove today. After they are done, there will be 21 trees. How many trees did the grove workers plant today?

A: There are 21 trees now and there are 15 trees in the beginning, so the workers plant $21 - 15 = 6$ trees. The answer is 6.

1. **Q:** If there are 3 cars in the parking lot and 2 more cars arrive, how many cars are in the parking lot?

A: There are 3 cars in the beginning, 2 more arrive, so now there should be $3 + 2 = 5$ cars. The answer is 5.

2. **Q:** Leah had 32 chocolates and her sister had 42. If they ate 35, how many pieces do they have left in total?

A: Leah had 32 chocolates and her sister had 42, in total they have $32 + 42 = 74$ chocolates. After they ate 35, now there are $74 - 35 = 39$ chocolates. The answer is 39.

3. **Q:** Jason had 20 lollipops. He gave Denny some lollipops. Now Jason has 12 lollipops. How many lollipops did Jason give to Denny?

A: Jason started with 20 lollipops, but now he only has 12, so he gave Denny $20 - 12 = 8$ lollipops. The answer is 8.

4. **Q:** Shawn has five toys. For Christmas, he got two toys each from his mom and dad. How many toys does he have now?

A: Shawn got 2 toys each from his mom and dad, so he got $2 * 2 = 4$ more, now he will have $5 + 4 = 9$ toys. The answer is 9.

5. **Q:** There were nine computers in the server room. Five more computers were installed each day, from monday to thursday. How many computers are now in the server room?

A: 5 computers were installed from monday to thursday, so in total $5 * 4 = 20$ computers are installed. 9 computers are there in the beginning, so now there are $20 + 9 = 29$ computers. The answer is 29.

6. **Q:** Michael had 58 golf balls. On tuesday, he lost 23 golf balls. On wednesday, he lost 2 more. How many golf balls did he have at the end of wednesday?

A: Michael started with 58 golf balls and lost 23, so he has $58 - 23 = 35$. After he lost 2 more, he has $35 - 2 = 33$ balls now. The answer is 33.

7. **Q:** Olivia has \$23. She bought five bagels for \$3 each. How much money does she have left?

A: 5 bagels for \$3 each should cost $5 * 3 = 15$ dollars. Olivia had \$23 in the beginning, so now she has $23 - 15 = 8$ dollars left. The answer is 8.