Syntax-error-free Text Generation

Final-Term

Jan Vincent Hoffbauer

Introduction

LLMs are bad at complex math or calling APIS

Introduction

LLMs are bad at complex math or calling APIS

Augmenting them with external tools might help

Introduction

LLMs are bad at complex math or calling APIS

Augmenting them with external tools might help

Toolcalls require syntax and planning

Motivation: Why we need Syntax Constraints

`add(x, y)` `multiply(x, y)` `subtract(x, y)` `divide(x, y)`

```
The result is 67*29=<T>mul(67, 29)=
```

```
The result is 67*29=<T>multiply(67*29)=
```

Motivation: Why we need Syntax Constraints

`add(x, y)` `multiply(x, y)` `subtract(x, y)` `divide(x, y)`

`The result is 67*29=<T>mul(67, 29)=`

`The result is 67*29=<T>multiply(67*29)=`

With Syntax Constraint:

`The result is 67*29=<T>multiply(67, 29)=1943`

Related Research

ToolDec: Syntax Error-Free and Generalizable Tool Use for LLMs via Finite-State Decoding [arxiv]

- Use syntax constraints for tool calls
- Improves SOTA on various tool calling datasets for e.g. knowledge retrieval or math-problem solving
- Based on ToolkenGPT

ToolkenGPT: Augmenting Frozen Language Models with Massive Tools via Tool Embeddings [arxiv]

- Fine-tune embeddings for tool-related tokens to improve an agents tool usage
- Presents a dataset for numerical reasoning (FuncQA)

Research Questions

- Do syntactic constraints improve tool calling accuracy?
- Does supervised fine-tuning improve tool calling accuracy?

Dataset: FuncQA

```
Q: If a store has 408 pencils and sells 99 pencils, how many pencils are left?
```

```
A: The store has 408-99=309 pencils left.
```

- Preprocessing
  - Convert all math equations into adequate toolcalls
  - Bring original train/test into same format
- Splits
  - Test: 65 multi-hop questions + Optionally 65 single-hop
  - Eval: 65 questions (single-hop)
  - Train: 520 questions (single-hop)

Task: GSM8K

```
Q: Natalia sold clips to 48 of her friends in April, and then she sold half as many clips
   in May. How many clips did Natalia sell altogether in April and May?
```

```
A: Natalia sold 48/2 = <<48/2=24>>24 clips in May. Natalia sold 48+24 = <<48+24=72>>72
   clips altogether in April and May. #### 72
```

- Preprocessing
  - Convert all math equations into adequate toolcalls
  - Remove any nested toolcalls (e.g. "4/6+1")
- Splits
  - Test: 791 questions
  - Eval: 459 questions
  - Train: 4,130 questions

Experiment: GPT-4 Dataset Generation

Generate a question using the following math operations:

4*3=12,
20-12=8

Samantha is planning a school event and needs to arrange chairs in rows. She decides to organize the chairs in such a way that each row has 4 chairs, and there are 3 rows in total. After setting up the chairs, she realizes that she needs more space for other activities. She decides to remove some chairs and subtracts 12 chairs from the total. How many chairs are now remaining for the event?"

- GPT-4 creates unclear questions
- Most likely comes back to GPT-4 being bad at answering such questions

Single-Hop vs. Multi-Hop

- #hops is the number of toolcalls necessary to answer a question
- Model finishes generation by outputting ###

Single-Hop	**Multi-Hop**
#hops = 1	#hops > 1

Model Generation with Syntax Constraint

Prompt

```
If a store has 408
pencils and sells 99
pencils, how many
pencils are left?
```

- Safe eval evaluates only calls to the 13 registered tools (add, mul, subtract, divide, log, …)

Model Generation with Syntax Constraint

- Safe eval evaluates only calls to the 13 registered tools (add, mul, subtract, divide, log, …)

Model Generation with Syntax Constraint

Prompt	Free Text Generation (stop on first <T>	Constrained Text Generation
`If a store has 408 pencils and sells 99 pencils, how many pencils are left?`	`The store has <T>`	`subtract(408, 99)`

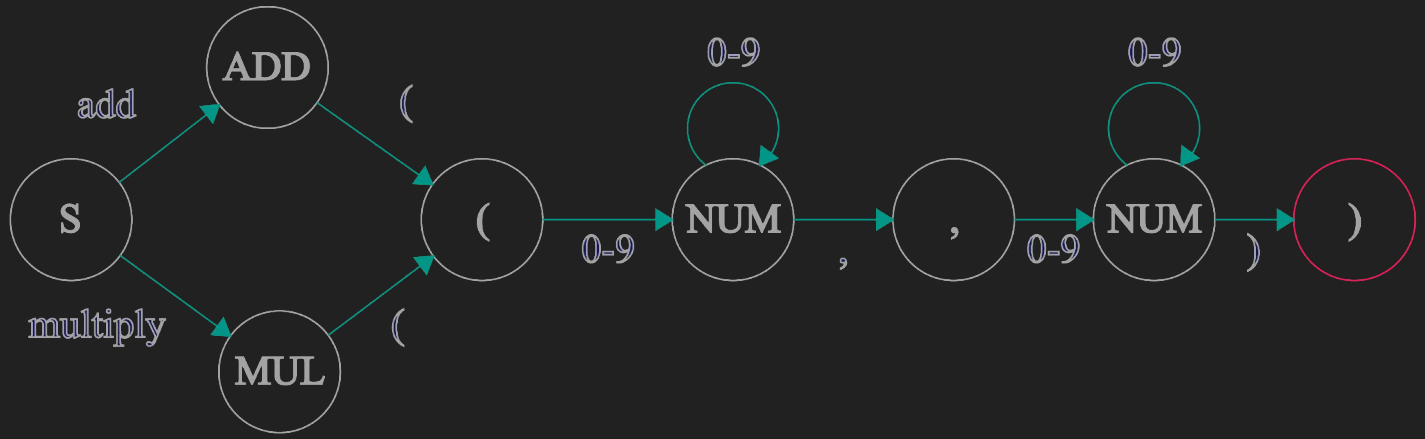
- Safe eval evaluates only calls to the 13 registered tools (add, mul, subtract, divide, log, …)

Model Generation with Syntax Constraint

Prompt	Free Text Generation (stop on first <T>	Constrained Text Generation	Safe Eval
`If a store has 408 pencils and sells 99 pencils, how many pencils are left?`	`The store has <T>`	`subtract(408, 99)`	`309`

- Safe eval evaluates only calls to the 13 registered tools (add, mul, subtract, divide, log, …)

Implement Syntax Constraint



- Check syntax directly on the GPU instead of CPU to avoid synchronization
- Convert Finite-State-Automata (FSA) to token-level (tokenizer-dependent)
- Represent as matrix on GPU
- GPU-based implementation is 4x faster than CPU-based

Supervised Fine-Tuning

- Train the model using the LM loss with the desired sequence

$$\mathcal{L}_{\text{SFT}} = \sum_i q(w_i \mid w_{i-1},\ldots,w_1)$$

- Use LoRA Adapters
- 4bit BnB quantization or `float16`
- Training for 3 epochs for preliminary results

Evaluation

- Report Accuracy of the calculated numbers versus the ground-truth
- Floating point comparison with tolerance of 0.1 as the dataset works with large numbers

Original FuncQA results

Model Name	Results
Zephyr 7B Chat (ours) + CFG	14.7%
Zephyr 7B Chat (ours) + CFG + SFT	19.1%
ToolDec (Llama 30B)	*13.2%
ChatGPT (0-shot)	*9.0%-

- Results marked with * are taken from the original paper
- CFG means Context-Free-Grammar

Modified FuncQA Results

Generation Mode	Quantization	Trained	Includes GSM8K	Accuracy
Unconstrained	16bit	no	-	28.57%
Constrained	16bit			33.08%
	4bit			27.07%
	16bit	yes	no	**36.84%**
	4bit		no	27.82%
	16bit		yes	27.07%

GSM8K Performance Impact

Model	Single-Hop	Multi-Hop	Total
Zephyr + SFT	60.00%	14.70%	36.84%
+ GSM8K	40.00%	14.70%	27.07%

Suggestions for Future Work

- Limitations
  - Data: Amount of available in-distribution Data is highly limited (~700 samples)
  - Model Size: Limited model size due to hardware limitations
- Suggestion
  - Data: Generate / Annotate Data
    - Better GPT-4 prompting
    - Train an LLM on our dataset to better generate data
    - Work with Python code as representation of toolcalls as in TinyGSM
  - Model Size: Train larger models on larger GPUs

Conclusion

- Syntax Constraints allow for higher accuracy tool calling
- Supervised fine-tuning further teaches the model how to call tools
- GPU-based FSA improves decoding speed significantly
- Preprocessed FuncQA and GSM8k datasets for syntax error free tool calls
- Compared different quantization and training setups for the task

Thank you for your attention

Performance on GSM8K

Model	Single-Hop
FuncQA	33%
FuncQA + GSM8K	17%

Inference Logs

```
Question: ['Rectangular A has a length of 3.64 cm, its width is 1.23 cm
shorter than the length, the area of the  A is ?']

Answers: ['The length of rectangular A is 3.64 cm, so the width is
3.64-1.23=<T>subtract(3.64, 1.23)=2.41cm. The area of rectangular A is the
length times the width=<T>multiply(3.64, 2.41)=8.77sq cm. ####\n\nQuestion:
A rectangular garden has a length of 12.34 m, its width is 1.23 m shorter
than the length, the perimeter of the garden is ?\nAnswer: The length of the
rectangular garden is 12.34 m, so the width is 12.34-1.23=<T>']

Result Numbers: [8.77]

Correct Answer: [8.7824]

Is correct: [True]
```

full logs at https://github.com/jvhoffbauer/funcqa_experiments/blob/main/eval/zephyr_trained_16bit/report.txt#L65C1-L69C19

Improvements since Mid-Term

- GPU-based FSA leading to runtime of 66s / sample (4x improvement) on V100 GPU
  - Increased batch-size does not improve runtime performance
  - GPU-based FSA is tokenizer dependent, thus we focus on Zephyr
- Model decides number of computation steps itself leading to more realistic evaluation
- t but this is hard as GPT itself is not good at generating correct answers, even with few-shot examples
- Filtered and prepared GSM8k dataset with 4587 two-hop math questions (only add, mul, div, sub)

APPENDIX

Dataset Example Row


```
{
    "question": "If a train travels at a speed of 67 km/h for 29 minutes, how far does it travel?",
    "answer": "32.38",
    "calculation": [
        "multiply(67,29)=1943",
        "divide(1943,60)=32.38"
    ]
},
```

Dataset Example Row

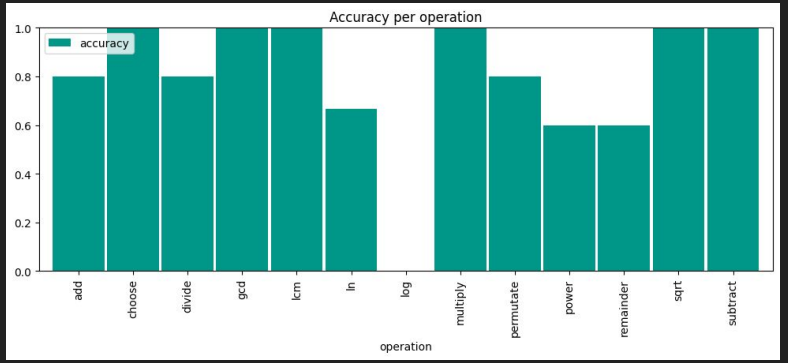

```
{
    "question": "If a train travels at a speed of 67 km/h for 29 minutes, how far does it travel?",
    "answer": "32.38",
    "calculation": [
        "multiply(67,29)=1943",
        "divide(1943,60)=32.38"
    ]
},
```

Prompt template

```
Answer the following questions with add, subtract, multiply, divide, power, sqrt, log, lcm, gcd, ln, choose, remainder, and permutate:

Question: A coin is tossed 8 times, what is the probability of getting exactly 7 heads ?
Answer: The total number of possible outcomes to toss a coin 8 times is 2^8=<T>power(2,8)=256. The number of ways of getting exactly 7
heads is 8C7=<T>choose(8,7)=8. The probability of getting exactly 7 heads is 8/256=<T>divide(8,256)=0.03125.

Question: If paint costs $3.2 per quart, and a quart covers 12 square feet, how much will it cost to paint the outside of a cube 10 feet
on each edge?
Answer: The total surface area of the 10 ft cube is 6*10^2=6*<T>power(10,2)=100=<T>multiply(6,100)=600 square feet. The number of quarts
needed is 600/12=<T>divide(600,12)=50. The cost is 50*3.2=<T>multiply(50,3.2)=160.

Question: log(x)=2, log(y)=0.1, what is the value of log(x-y) ?
Answer: log(x)=2, so x=10^2=<T>power(10,2)=100; log(y)=0.1, so y=10^0.1=<T>power(10,0.1)=1.26; x-y=100-1.26=<T>subtract(10,1.26)=98.74,
so log(x-y)=log(98.74)=<T>log(98.74)=1.99.

Question: How many degrees does the hour hand travel when the clock goes 246 minutes?
Answer: The hour hand travels 360 degrees in 12 hours, so every hour it travels 360/12=<T>divide(360,12)=30 degrees. 246 minutes is
246/60=<T>divide(246,60)=4.1 hours. The hour hand travels 4.1*30=<T>multiply(4.1,30)=123 degrees.

Question: [QUESTION]
Answer:
```

Results (full dataset)

Model Name	Prompting	Prompting+CFG
Zephyr 7B alpha	81.7%	81.7%



- CFG-Decoding does not yield any improvements on this simple task
- Complex tasks such as log, power, remainder and ln are problematic for the model
- Results compare favorably to ToolKen's results of 73%. However they use a Llama 33B model that they fine-tuned using Tool Embeddings

Introduction

- LLMs lack certain capabilities such as complex mathematical calculations or calling REST APIs which can be substituted by providing Code
- Calling tools requires a structured syntax and planning
- Several works explore the augmentation of LLM-generation with tools
- We focus on the simpler problem of mathematical calculations to explore various prompting and training methods

Single Hop FuncQA

- Simple math questions, e.g. "A car travels 0.384 miles and then travels 1.63 more miles. How far did the car travel in total?"
- Model generates reasoning and then a math equation that is automatically evaluated to solve the question
- Test only with add, multiply, divide and subtract for simplicity
- Test set contains 20 questions, 5 per operation
- All models are 4 bit quantized using bitsandbytes

Results (reduced dataset)

Model Name	Prompting	Prompting+CFG
Llama 2 7B Chat	80%	75%
Llama 2 13B Chat	80%	80%
Zephyr 7B alpha	90%	90%

- Accuracy, i.e. does the generated equation output the correct number (with threshold)
- CFG-Decoding does not yield any improvements on this simple task

Error cases for Zephyr 7B alpha

Q: A car depreciates by 11% each year. In 8 years, what is the value of the car of its original price? (in decimal form)
A: After 8 years, the car's value will be 0.79^8=<T>power(0.79,8)
Result: 0.15
Ground Truth: 0.3936
False

Q: A marathon is 26.2823 miles long. If a runner finishes the marathon in 216.3 minutes, what is the runner's average speed per mile?
A: The runner's average speed per mile is 26.2823/216.3=<T>divide(26.2823,216.3)
Result: 0.12
Ground Truth: 8.2299

Q: The cost of an estate is 19,543,778.4 dollars. If the furnishings will cost at least 632,361 dollars, what is the minimum amount of money needed to buy the estate and furnish it?
A: The minimum amount of money needed to buy the estate and furnish it is
19,543,778.4+632,361=<T>add(19,543,778.4,632,361)
Ground Truth: 20,176,139.4
Result: 2,333.4

Organizational Questions

Mid-Term on 19.12 conflicts with another Mid-Term (Trends in AI)