UNIVERSITY of WASHINGTON

Teaching Language Models to Use External Tools

Emmitt Choi, Katelyn Mei, Bruno Zorrilla

PAUL G. ALLEN SCHOOL
OF COMPUTER SCIENCE & ENGINEERING

Motivation

- Current state-of-the-art large language models (LLMs) are limited in certain downstream tasks such as mathematical calculation despite their human-like performance in natural language generation.
- One potential solution to overcome these limitations is enabling language models to utilize existing tools.

Research Question:

Can we teach GPT-J to use external tools like calculators and Wikipedia search?

Methodology

- **Datasets**: ASDiv (2.4k), Natural Questions (2.4k) , dataset annotated w/ API calls. 80-20 split between train (finetuning) & test (benchmark)
- **Model:** GPT-J (6B)
- **Training Procedure:** Finetune on 5 epochs on 3.6k math+Q&A datapoints, for both vanilla and Toolformer

Our Toolformer model is GPT-J finetuned on a dataset annotated w/ API calls:

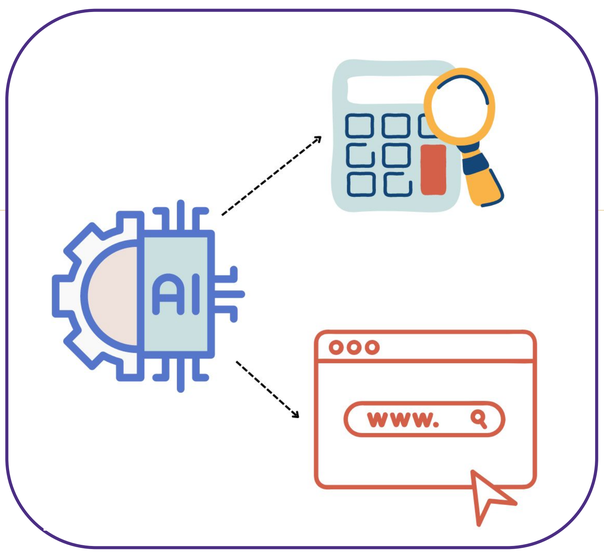
Unlike the original paper (where toolformer training was end-to-end on the GPT-J model), we had GPT-3.5 assist with choosing a tool API for each datapoint, and adding said API annotation for said datapoint.

Calculator() Annotation example:

Preannotated: We have 4 * 30 ft = 120 ft.
Annotated: We have 4 * 30 ft = [Calculator(4 * 30)->120] 120 ft.

WikiSearch() Annotation example:

Preannotated: What's the sky's color?
Annotated: What's the sky's color? [WikiSearch("sky's color")-> "Most of the light in the sky is caused by..."]



Results

Math benchmark

In math, our Toolformer does a **bit better** than vanilla model!

45.8% vs **43.6%**

Toolformer math acc | Vanilla math acc

This performance boost may be attributed to Toolformer's calculator use for arithmetic, which allows the LM's errors due to miscalculations vanish to 0.

Q&A benchmark

Unfortunately, in Q&A, our Toolformer gets **nothing** correct! 😰

0% vs **2.67%**

Toolformer Q&A acc | Vanilla Q&A acc

Toolformer learns to search Wikipedia, but doesn't ever answer the Q&A questions with WikiP's info, & only answers w/ runaway tangents...

Conclusion

- We're able to partially replicate the original paper, by building a self-supervised model capable of benefiting in math w/ calculator use.
- However, unlike the original paper, our Toolformer was not able to benefit from retrieving Wikipedia snippets whatsoever.

Future Work

- Use GPT-J to annotate the training set itself with APIs, based off the tools chosen as optimal by GPT-4, thus lowering computational costs.
- Add additional tool functionality, for things like Calendar() and Translate().
- Make our Toolformer model learn the most optimal tools to use, by having it optimize under the computed loss of its own generated annotations.

Reference

Schick, Timo, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda and Thomas Scialom. "Toolformer: Language Models Can Teach Themselves to Use Tools." ArXiv abs/2302.04761 (2023): n. pag.