Toolformer: Language Models Can Teach Themselves to Use Tools

건설환경공학과 21학번 이현

https://arxiv.org/abs/2302.04761

CONTENTS

00

Toolformer란?

01

Introduction

- LM 한계점 & 해결책
- Toolformer의 주요목표

02

Approach

- API call
- API call 주요과정

03

Experiment

- Baseline Model
- Downstream Tasks
- Scaling Laws

04

Analysis

- Decoding Strategy
- Data Quality

05

Limitations

06

Conclustion

Chapter
00

Toolformer 란?

Toolformer : 어떤 API를 호출할지, 언제 호출할지, 어떤 arguments를 전달할지, 결과를 미래 토큰 예측에 어떻게 최선으로 통합할지를 결정하는 모델

- calculator
- Q&A system
- search engine
- translation
- calendar

Downstream Task에서 zero-shot 성능 향상!!!
More large LM과의 경쟁력 보유!!!

Introduction

LM의 한계점

- 산술 연산
- 사실 조회

한계점의 해결책

API call을 통한 LM의 강점과
외부 Tool의 결합

Introduction

Toolformer의 주요 목표

- Tool의 사용은 human annotations 가 아닌 self-supervised 방식으로 학습되어야 한다.
- LM은 genearlity를 잃지 않고, 스스로 언제 어떤 도구를 사용할지 결정할 수 있어야 한다.

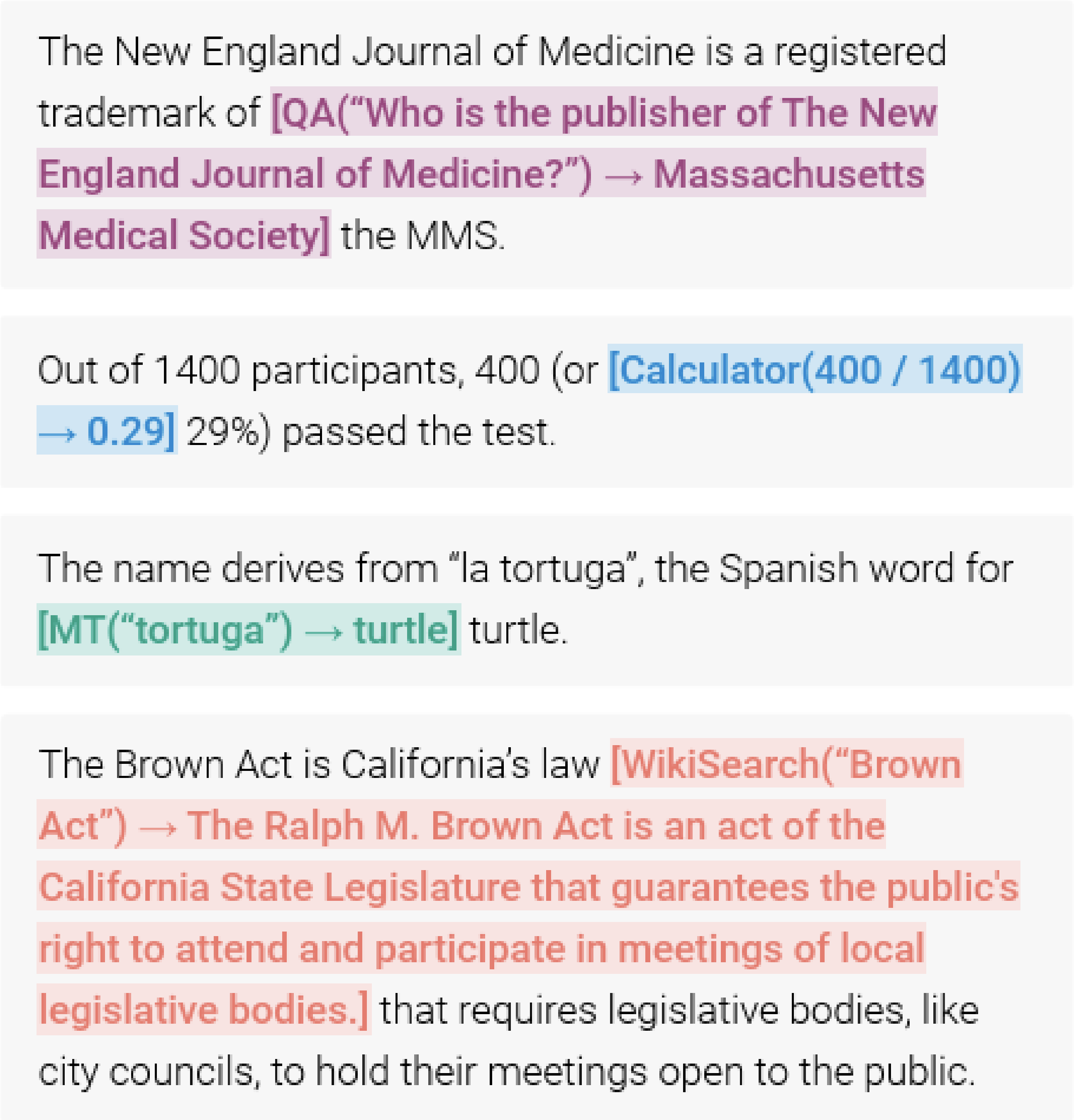
GPT-J 모델을 사전학습시키고, 6.78B parameters를 가진 것이
더 큰 GPT-3 model의 성능을 넘어선다.

Approach

API call

$$e(c) = \text{<API>}\, a_c\,(i_c)\ \text{</API>}$$
$$e(c, r) = \text{<API>}\, a_c\,(i_c)\ \rightarrow r\ \text{</API>}$$

- tuple c = (a_c, i_c)
- a_c : API 명
- i_c : input
- r : API call 결과



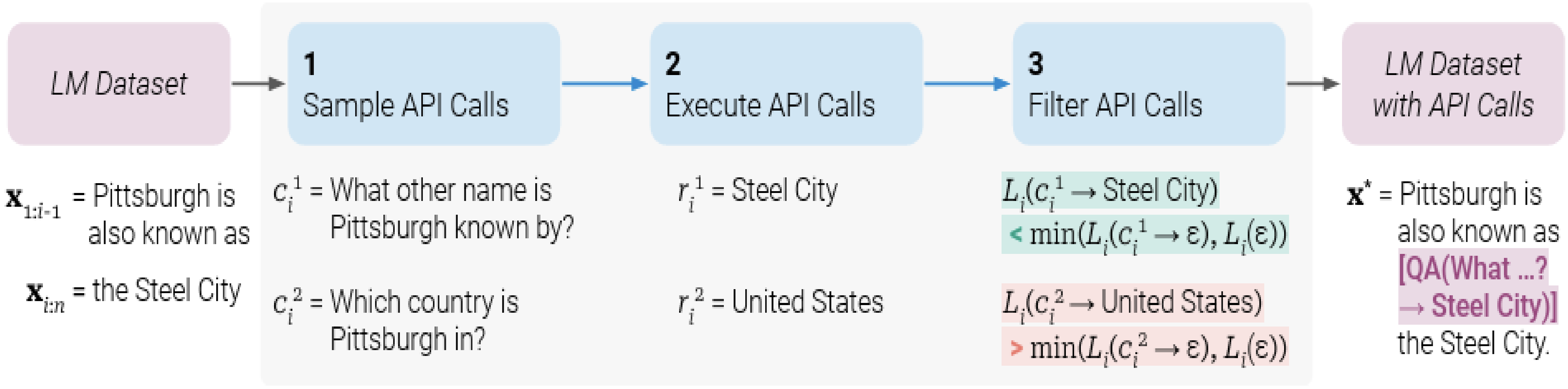
The New England Journal of Medicine is a registered trademark of [QA("Who is the publisher of The New England Journal of Medicine?") → Massachusetts Medical Society] the MMS.

Out of 1400 participants, 400 (or [Calculator(400 / 1400) → 0.29] 29%) passed the test.

The name derives from "la tortuga", the Spanish word for [MT("tortuga") → turtle] turtle.

The Brown Act is California's law [WikiSearch("Brown Act") → The Ralph M. Brown Act is an act of the California State Legislature that guarantees the public's right to attend and participate in meetings of local legislative bodies.] that requires legislative bodies, like city councils, to hold their meetings open to the public.

Dataset에서 API call 과정



$\mathbf{x}_{1:i-1}$ = Pittsburgh is also known as

$\mathbf{x}_{i:n}$ = the Steel City

c_i^1 = What other name is Pittsburgh known by?

c_i^2 = Which country is Pittsburgh in?

r_i^1 = Steel City

r_i^2 = United States

$L_i(c_i^1 \rightarrow \text{Steel City})$
$< \min(L_i(c_i^1 \rightarrow \varepsilon), L_i(\varepsilon))$

$L_i(c_i^2 \rightarrow \text{United States})$
$> \min(L_i(c_i^2 \rightarrow \varepsilon), L_i(\varepsilon))$

$\mathbf{x}^*$ = Pittsburgh is also known as [QA(What ...? → Steel City)] the Steel City.

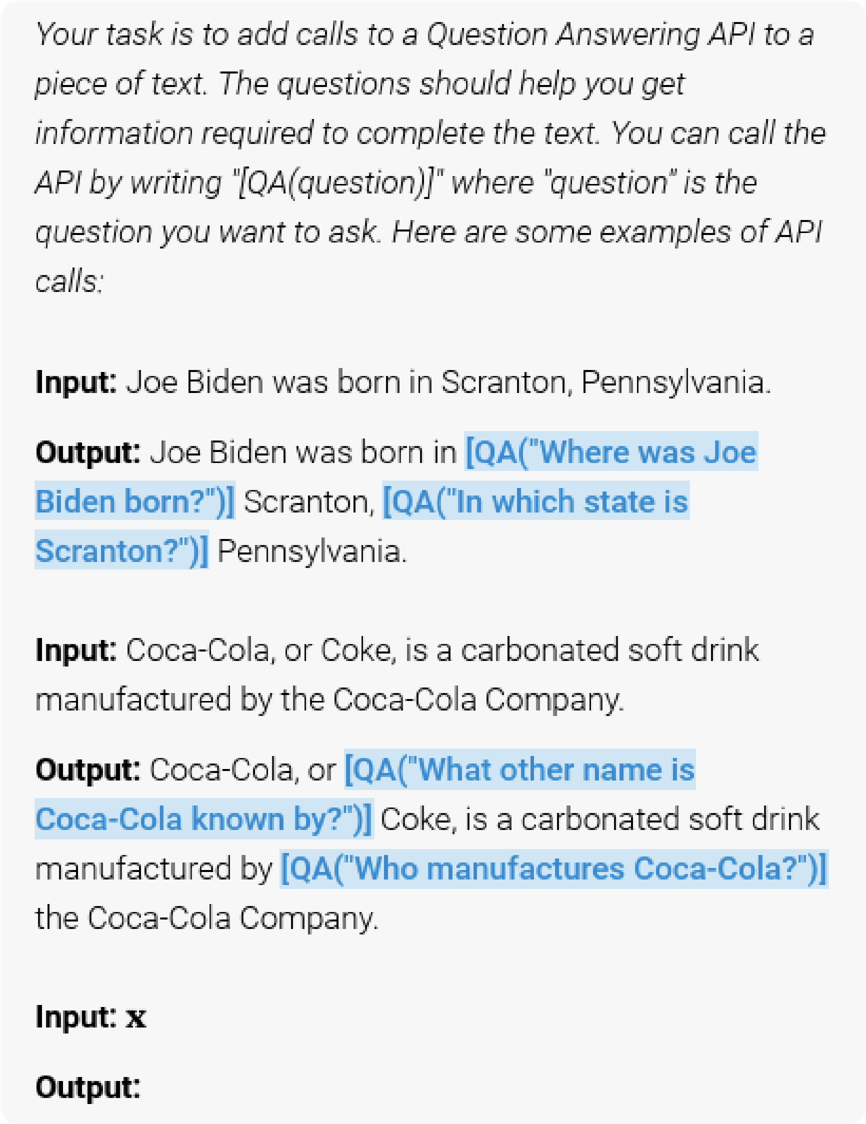
Chapter
02

Approach

Sampling API Calls

Sampling 식

$$p_i = p_M(\texttt{<API>} \mid P(\mathbf{x}), x_{1:i-1})$$



Your task is to add calls to a Question Answering API to a piece of text. The questions should help you get information required to complete the text. You can call the API by writing "[QA(question)]" where "question" is the question you want to ask. Here are some examples of API calls:

Input: Joe Biden was born in Scranton, Pennsylvania.

Output: Joe Biden was born in **[QA("Where was Joe Biden born?")]** Scranton, **[QA("In which state is Scranton?")]** Pennsylvania.

Input: Coca-Cola, or Coke, is a carbonated soft drink manufactured by the Coca-Cola Company.

Output: Coca-Cola, or **[QA("What other name is Coca-Cola known by?")]** Coke, is a carbonated soft drink manufactured by **[QA("Who manufactures Coca-Cola?")]** the Coca-Cola Company.

Input: $\mathbf{x}$

Output:

generate API calls를 QA에서 사용한 Prompt P(x)

Executing API Calls & Filtering API Calls

M에 대한 weighted cross entropy Loss

$$L_i(\mathbf{z}) = -\sum_{j=i}^{n} w_{j-i} \cdot \log p_M(x_j \mid \mathbf{z}, x_{1:j-1})$$

Z 접두사를 따름

$$L_i^+ = L_i(\mathrm{e}(c_i, r_i))$$
$$L_i^- = \min\left(L_i(\varepsilon), L_i(\mathrm{e}(c_i, \varepsilon))\right)$$

$$L_i^- - L_i^+ \geq \tau_f$$

Li_+ : API call과 그 결과가 M에 접두사로 주어졌을 때, 모든 토큰 x_i, ..., x_n에 대한 weighted loss
Li_- : API call을 전혀 하지 않거나 API call은 하지만, 응답을 제공하지 않는 경우 loss의 최솟값

r_f : 필터링 임계값

Model Fine-tuning & Inference

Fine-tuning

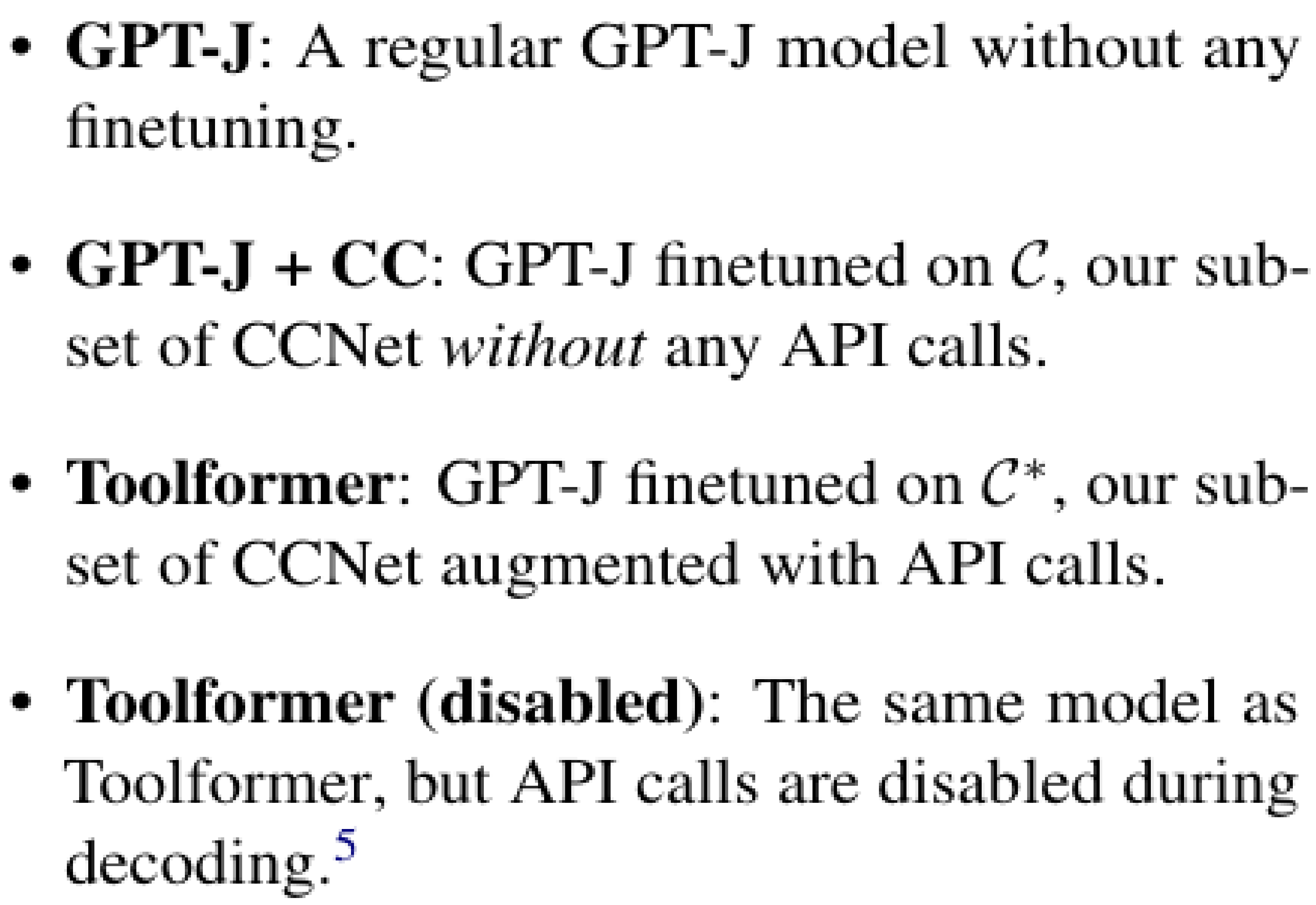
- API call이 추가된 새로운 데이터 셋 C*
- C*로 M을 fine-tuning

API call이 정확히 M이 미래 토큰을 예측하는데 도움이 되는 위치와 입력에 삽입되어 자체적인 피드백 기반이 형성됨

Inference

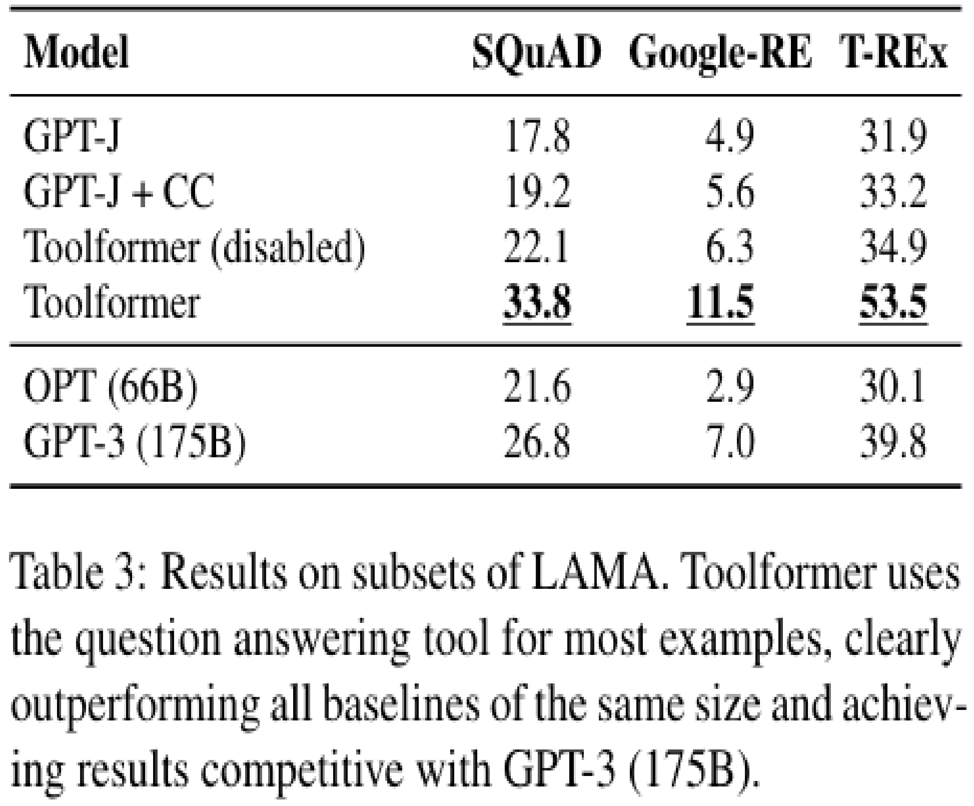
"→" 토큰을 생성할 때까지 정규 디코딩 수행 → "→"는 API call에 대한 응답 기대
응답과 </API> 토큰 삽입 후 디코딩 과정 지속

Baseline Model



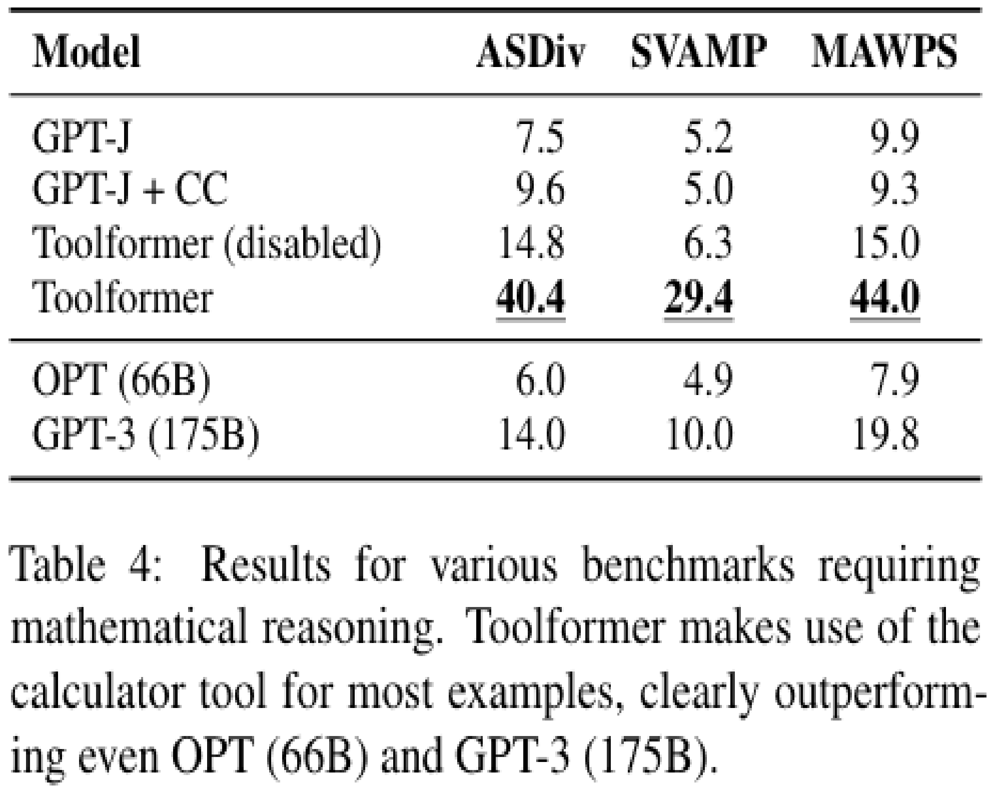
- **GPT-J**: A regular GPT-J model without any finetuning.
- **GPT-J + CC**: GPT-J finetuned on $\mathcal{C}$, our subset of CCNet *without* any API calls.
- **Toolformer**: GPT-J finetuned on $\mathcal{C}^*$, our subset of CCNet augmented with API calls.
- **Toolformer (disabled)**: The same model as Toolformer, but API calls are disabled during decoding.[5]

Downstream Tasks



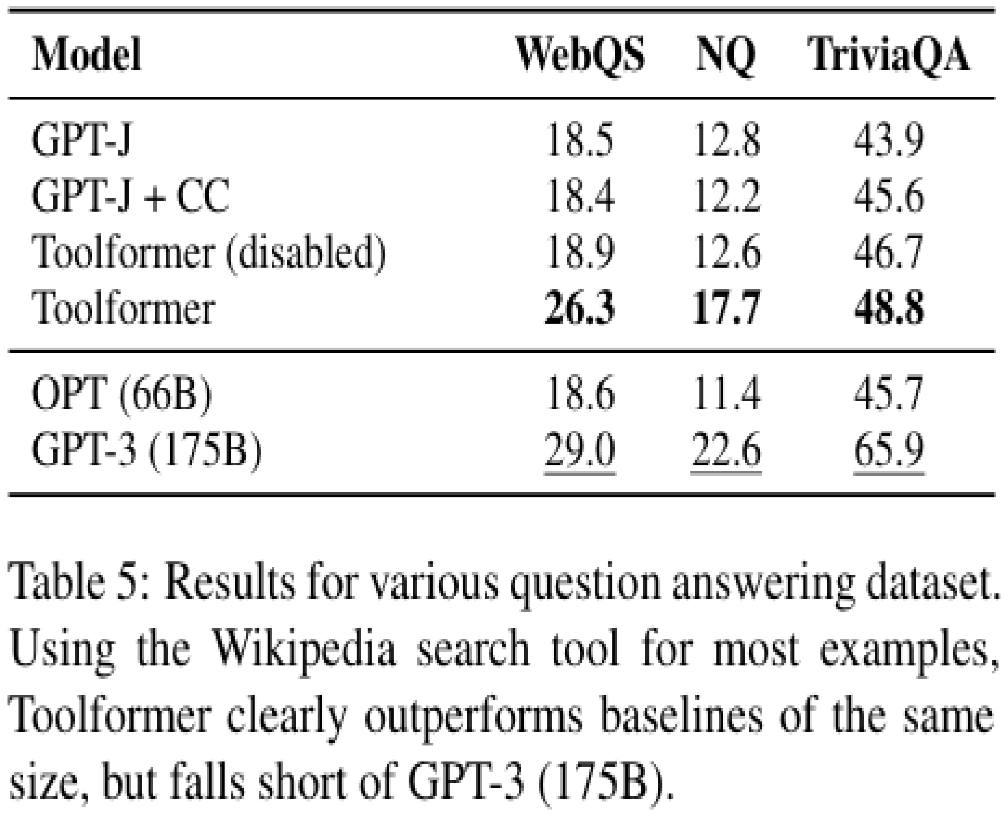
Model	SQuAD	Google-RE	T-REx
GPT-J	17.8	4.9	31.9
GPT-J + CC	19.2	5.6	33.2
Toolformer (disabled)	22.1	6.3	34.9
Toolformer	**33.8**	**11.5**	**53.5**
OPT (66B)	21.6	2.9	30.1
GPT-3 (175B)	26.8	7.0	39.8

Table 3: Results on subsets of LAMA. Toolformer uses the question answering tool for most examples, clearly outperforming all baselines of the same size and achieving results competitive with GPT-3 (175B).



Model	ASDiv	SVAMP	MAWPS
GPT-J	7.5	5.2	9.9
GPT-J + CC	9.6	5.0	9.3
Toolformer (disabled)	14.8	6.3	15.0
Toolformer	**40.4**	**29.4**	**44.0**
OPT (66B)	6.0	4.9	7.9
GPT-3 (175B)	14.0	10.0	19.8

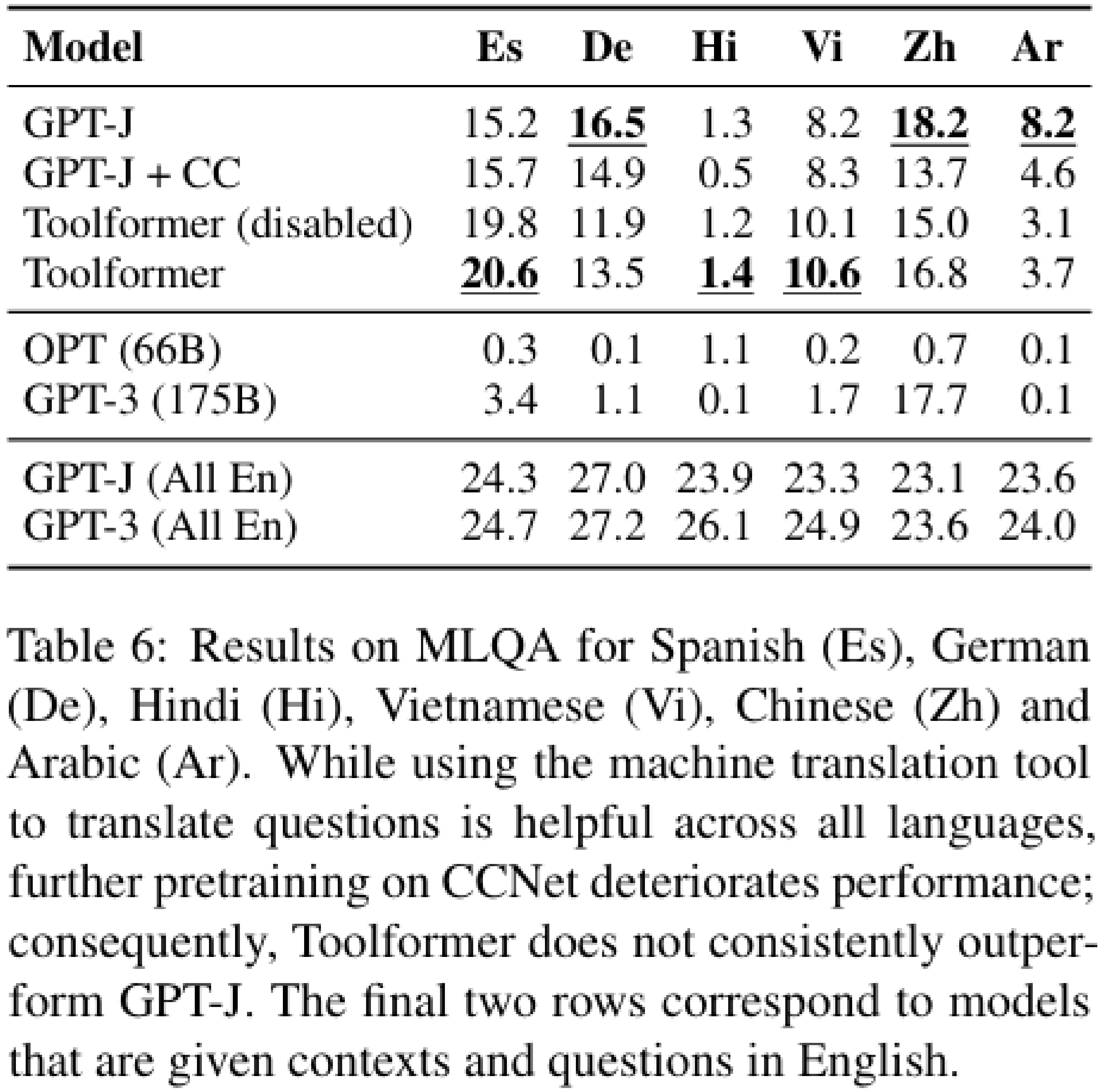
Table 4: Results for various benchmarks requiring mathematical reasoning. Toolformer makes use of the calculator tool for most examples, clearly outperforming even OPT (66B) and GPT-3 (175B).



Model	WebQS	NQ	TriviaQA
GPT-J	18.5	12.8	43.9
GPT-J + CC	18.4	12.2	45.6
Toolformer (disabled)	18.9	12.6	46.7
Toolformer	**26.3**	**17.7**	**48.8**
OPT (66B)	18.6	11.4	45.7
GPT-3 (175B)	29.0	22.6	65.9

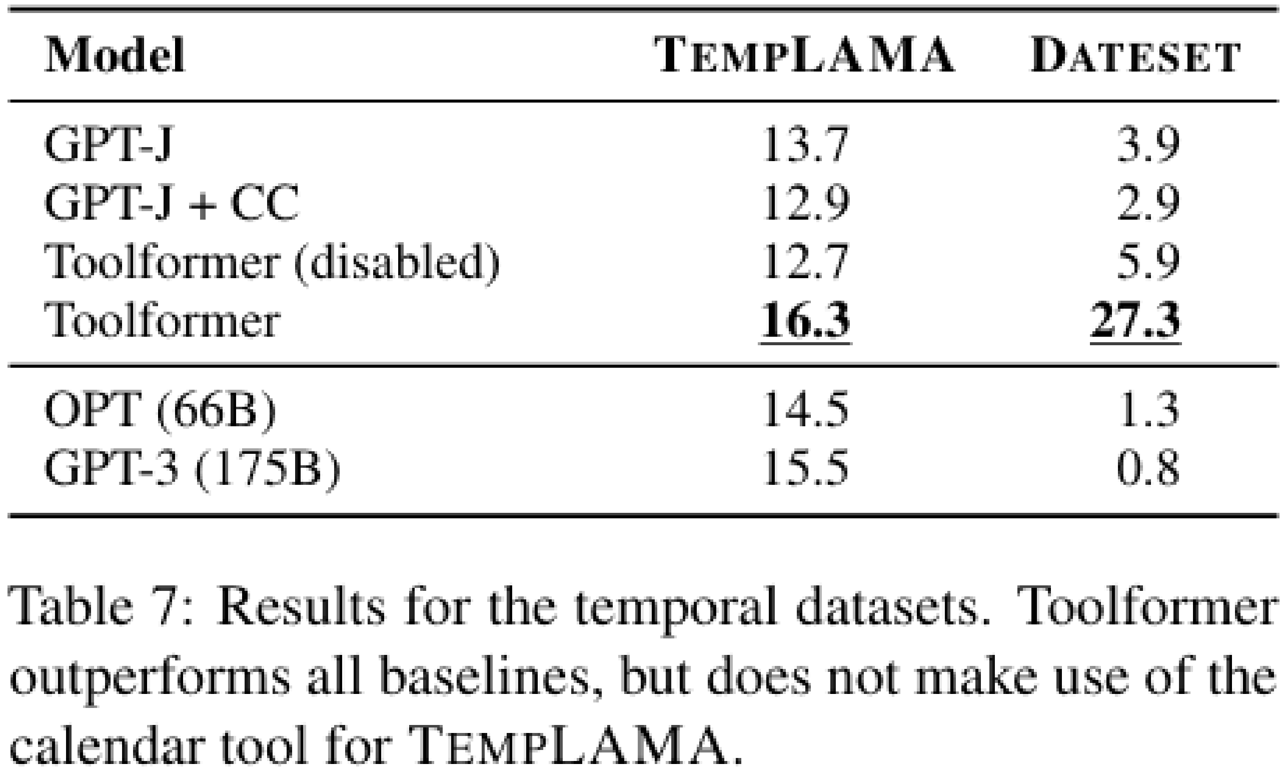
Table 5: Results for various question answering dataset. Using the Wikipedia search tool for most examples, Toolformer clearly outperforms baselines of the same size, but falls short of GPT-3 (175B).

Downstream Tasks



Model	Es	De	Hi	Vi	Zh	Ar
GPT-J	15.2	**16.5**	1.3	8.2	**18.2**	**8.2**
GPT-J + CC	15.7	14.9	0.5	8.3	13.7	4.6
Toolformer (disabled)	19.8	11.9	1.2	10.1	15.0	3.1
Toolformer	**20.6**	13.5	**1.4**	**10.6**	16.8	3.7
OPT (66B)	0.3	0.1	1.1	0.2	0.7	0.1
GPT-3 (175B)	3.4	1.1	0.1	1.7	17.7	0.1
GPT-J (All En)	24.3	27.0	23.9	23.3	23.1	23.6
GPT-3 (All En)	24.7	27.2	26.1	24.9	23.6	24.0

Table 6: Results on MLQA for Spanish (Es), German (De), Hindi (Hi), Vietnamese (Vi), Chinese (Zh) and Arabic (Ar). While using the machine translation tool to translate questions is helpful across all languages, further pretraining on CCNet deteriorates performance; consequently, Toolformer does not consistently outperform GPT-J. The final two rows correspond to models that are given contexts and questions in English.



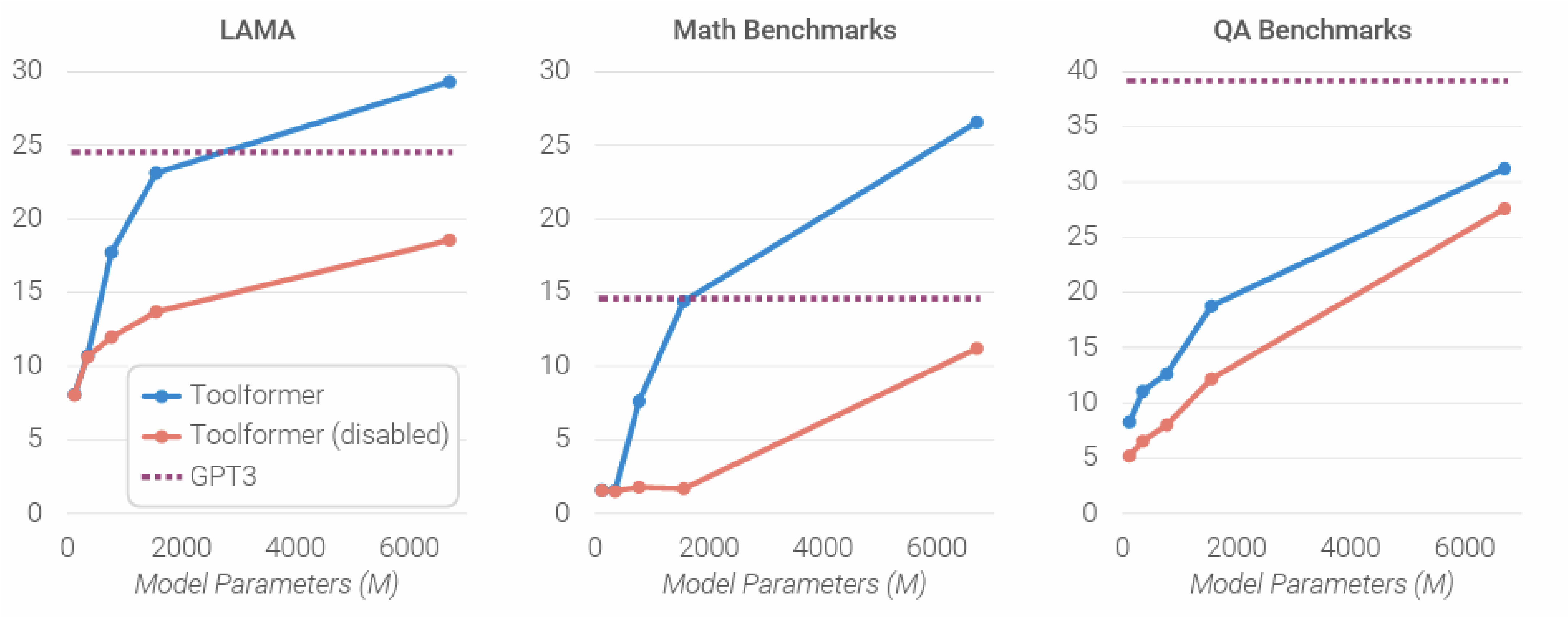
Model	TEMPLAMA	DATESET
GPT-J	13.7	3.9
GPT-J + CC	12.9	2.9
Toolformer (disabled)	12.7	5.9
Toolformer	**16.3**	**27.3**
OPT (66B)	14.5	1.3
GPT-3 (175B)	15.5	0.8

Table 7: Results for the temporal datasets. Toolformer outperforms all baselines, but does not make use of the calendar tool for TEMPLAMA.

Chapter
03

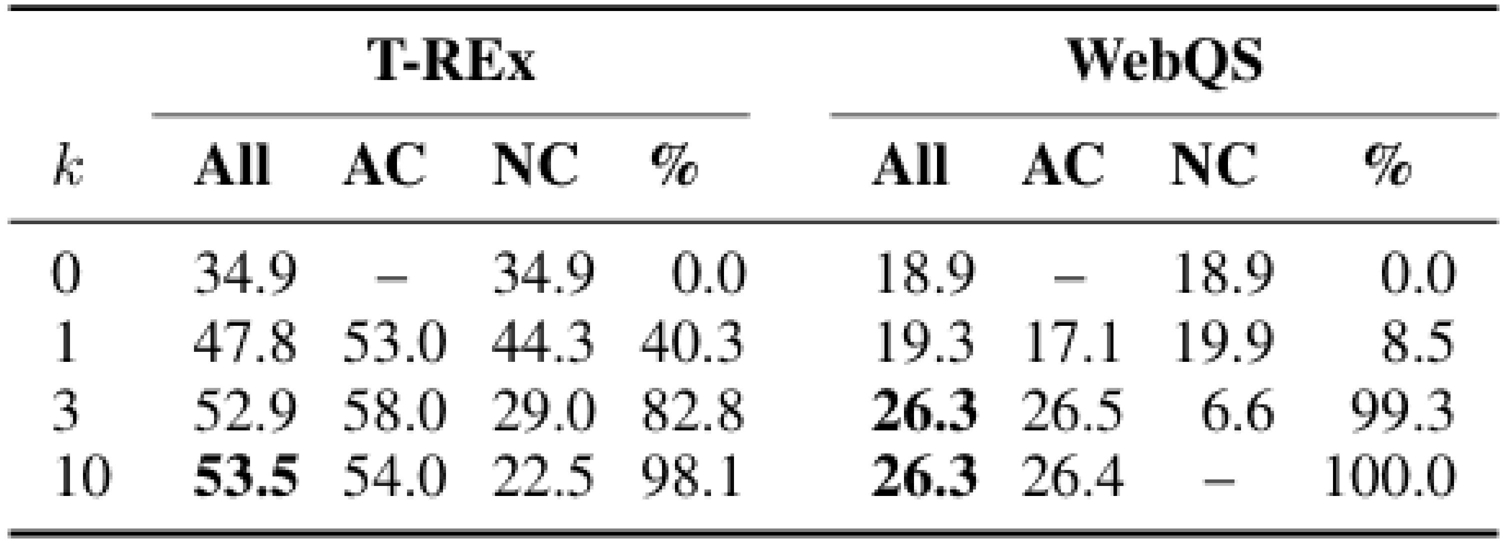
Experments

Scaling Laws



- Tool을 제대로 사용할 수 있는 능력은 775M parameter 부터 시작
- 모델이 커질 수록 API call 활용 능력도 커짐

Decoding Strategy



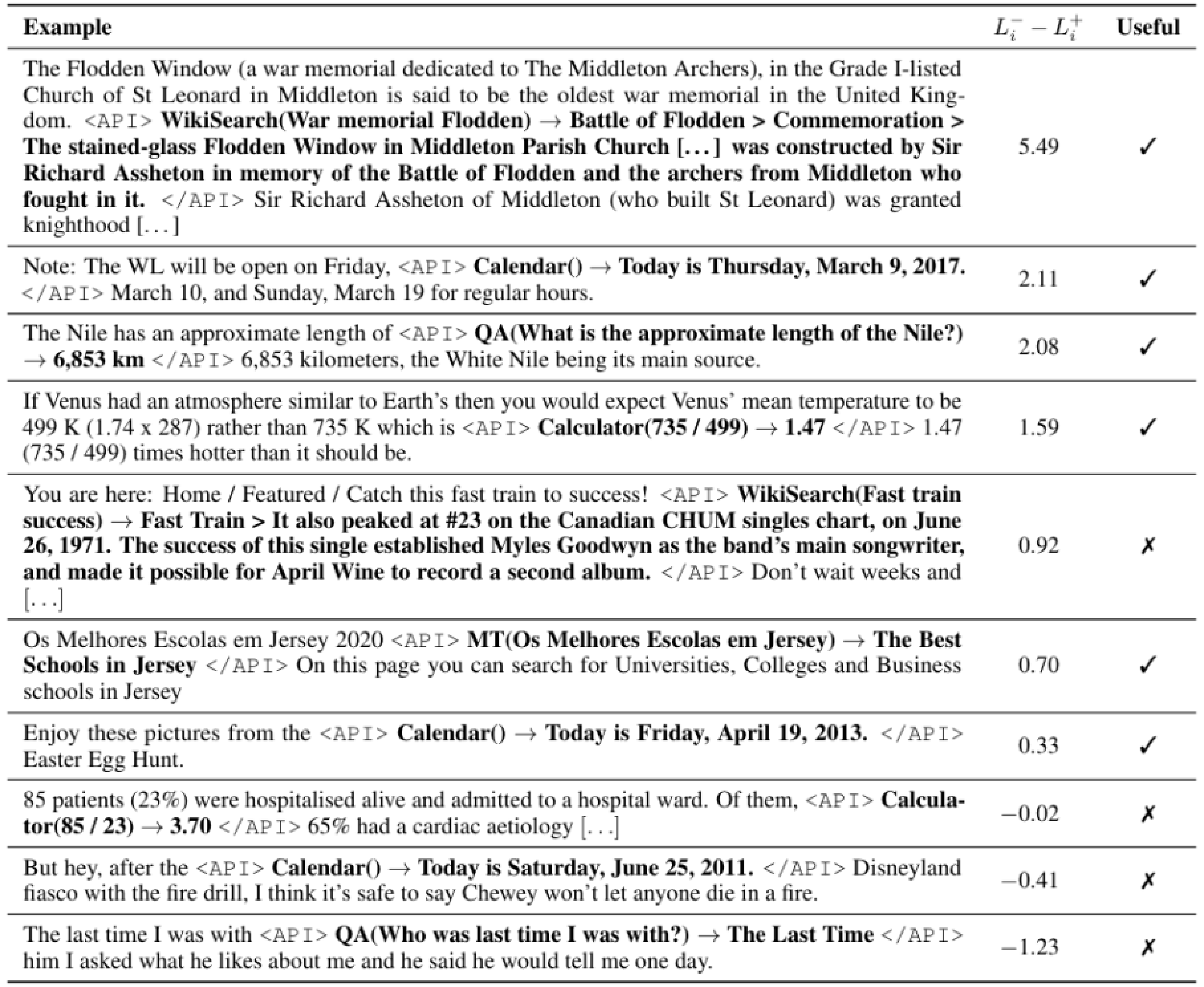
	T-REx				WebQS			
k	**All**	**AC**	**NC**	**%**	**All**	**AC**	**NC**	**%**
0	34.9	–	34.9	0.0	18.9	–	18.9	0.0
1	47.8	53.0	44.3	40.3	19.3	17.1	19.9	8.5
3	52.9	58.0	29.0	82.8	**26.3**	26.5	6.6	99.3
10	**53.5**	54.0	22.5	98.1	**26.3**	26.4	–	100.0

- k 증가에 따라 더 많은 예제에서 API call을 수행한다.
- k = 1 vs k = 10
- WebQS : k = 1-> k = 3 을 통해 엄청난 API call 발견 가능

Chapter
04

Analysis

Data Quality



Example	$L_i^- - L_i^+$	Useful
The Flodden Window (a war memorial dedicated to The Middleton Archers), in the Grade I-listed Church of St Leonard in Middleton is said to be the oldest war memorial in the United Kingdom. <API> **WikiSearch(War memorial Flodden) → Battle of Flodden > Commemoration > The stained-glass Flodden Window in Middleton Parish Church [. . .] was constructed by Sir Richard Assheton in memory of the Battle of Flodden and the archers from Middleton who fought in it.** </API> Sir Richard Assheton of Middleton (who built St Leonard) was granted knighthood [. . .]	5.49	✓
Note: The WL will be open on Friday, <API> **Calendar() → Today is Thursday, March 9, 2017.** </API> March 10, and Sunday, March 19 for regular hours.	2.11	✓
The Nile has an approximate length of <API> **QA(What is the approximate length of the Nile?) → 6,853 km** </API> 6,853 kilometers, the White Nile being its main source.	2.08	✓
If Venus had an atmosphere similar to Earth's then you would expect Venus' mean temperature to be 499 K (1.74 x 287) rather than 735 K which is <API> **Calculator(735 / 499) → 1.47** </API> 1.47 (735 / 499) times hotter than it should be.	1.59	✓
You are here: Home / Featured / Catch this fast train to success! <API> **WikiSearch(Fast train success) → Fast Train > It also peaked at #23 on the Canadian CHUM singles chart, on June 26, 1971. The success of this single established Myles Goodwyn as the band's main songwriter, and made it possible for April Wine to record a second album.** </API> Don't wait weeks and [. . .]	0.92	✗
Os Melhores Escolas em Jersey 2020 <API> **MT(Os Melhores Escolas em Jersey) → The Best Schools in Jersey** </API> On this page you can search for Universities, Colleges and Business schools in Jersey	0.70	✓
Enjoy these pictures from the <API> **Calendar() → Today is Friday, April 19, 2013.** </API> Easter Egg Hunt.	0.33	✓
85 patients (23%) were hospitalised alive and admitted to a hospital ward. Of them, <API> **Calculator(85 / 23) → 3.70** </API> 65% had a cardiac aetiology [. . .]	−0.02	✗
But hey, after the <API> **Calendar() → Today is Saturday, June 25, 2011.** </API> Disneyland fiasco with the fire drill, I think it's safe to say Chewey won't let anyone die in a fire.	−0.41	✗
The last time I was with <API> **QA(Who was last time I was with?) → The Last Time** </API> him I asked what he likes about me and he said he would tell me one day.	−1.23	✗

Table 10: Examples of API calls for different tools, sorted by the value of $L_i^- - L_i^+$ that is used as a filtering criterion. High values typically correspond to API calls that are intuitively useful for predicting future tokens.

Li_(-) - Li_(+) score 로 유용한 API call 판정

Limitations

1. Toolformer가 Tool을 연쇄적으로 사용할 능력이 없다는 것.
2. LM이 Tool을 상호작용적으로 사용할 수 없다는 것.
3. API call을 결정할 때 input의 표현에 민감하다. (프롬프트에 LM이 민감하기 때문)
4. API call을 결정할 때, 현재 API call로 인해 발생하는 Tool 별 cost를 고려하지 않는다.

Conclusion

1. API call 만으로 검색 엔진, 달력, 번역 시스템을 LM이 self-supervised 방식으로 학습하는 Toolformer

2. API call을 단순한 fine-tuning으로 해결 가능

3. 6.78B GPT-J 모델의 zero-shot 성능 향상

4. 다양한 Downstream Task에서 GPT-3 모델 성능 능가

감사합니다