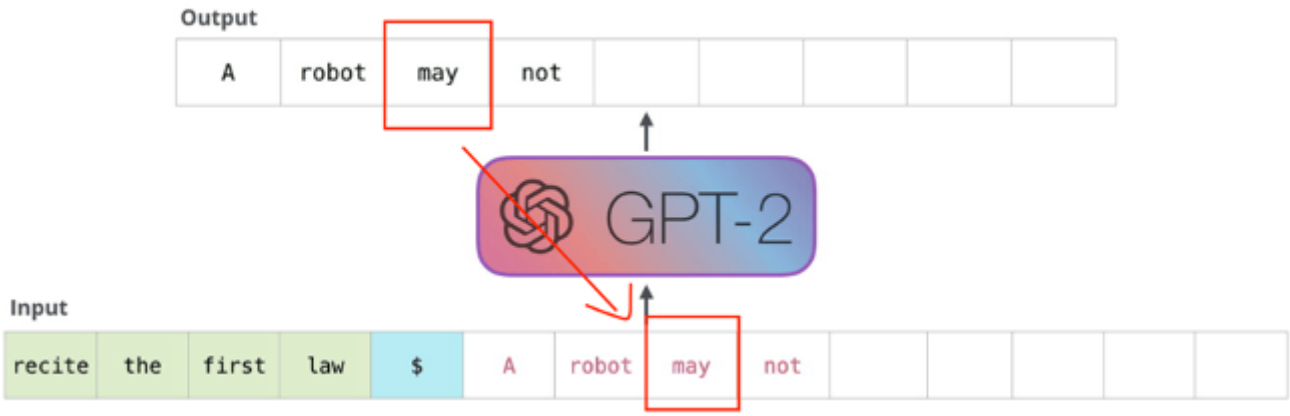
[GPT-2] Language Models are Unsupervised Multitask Learners

1. 논문이 다루는 Task



Task: Text Generation

- Input: Text
- Output:Text
- Text Generation : 단순한 텍스트 생성 모델이다. Auto Regressive하게 이전의 input이 다음의 input으로 들어가는 모델이다.

2. 기존 연구 한계

2-1. Domain-Specific Training

기존의 딥러닝에서는 Domain에 맞게 파인튜닝하는 과정을 거쳤다. GPT-1 또한 파인튜닝하는 과정을 거쳤어야 하며 그 후에 나온 BERT 모델 또한 Layer를 수정하며 파인튜닝을 진행해야한다. 이는 전체적인 딥러닝의 일반화를 제한한다고 주장하며 보다 광범위하게 즉 Zero-shot을 해야한다고 주장한다.

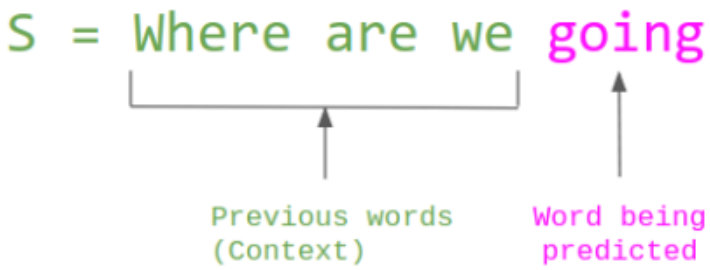
2-2. 기존 ML모델의 교육

기존 ML 시스템에서의 특정 테스크를 수행하기 위해서는 테스크에 맞는 데이터 수천 수백개의 예제가 필요하다. 하지만 이는 매우 좁은 범위이며 모든 테스크와 도메인에 대해 데이터 쌍을 만드는 것은 상당히 어려울 것이다. 그래서 전체적인 일반화를 높히기 다중 학습(Multitask Learners)가 효과적인좋은 선택이라고 주장한다.

3. 제안 방법론

3-1. Language Modeling

$$P_{(w_1,w_2,...,w_n)} = p(w_1)p(w_2|w_1)p(w_3|w_1,w_2)...p(w_n|w_1,w_2,..,w_{n-1}) = \prod_{i=1}^{n} p(w_i|w_1,...,w_{i-1}) \quad (1)$$



P(S) = P(Where) x P(are | Where) x P(we | Where are) x P(going | Where are we)

문장의 확률은 이전 기호가 주어진 각 기호의 확률의 곱으로 정의할 수 있습니다.

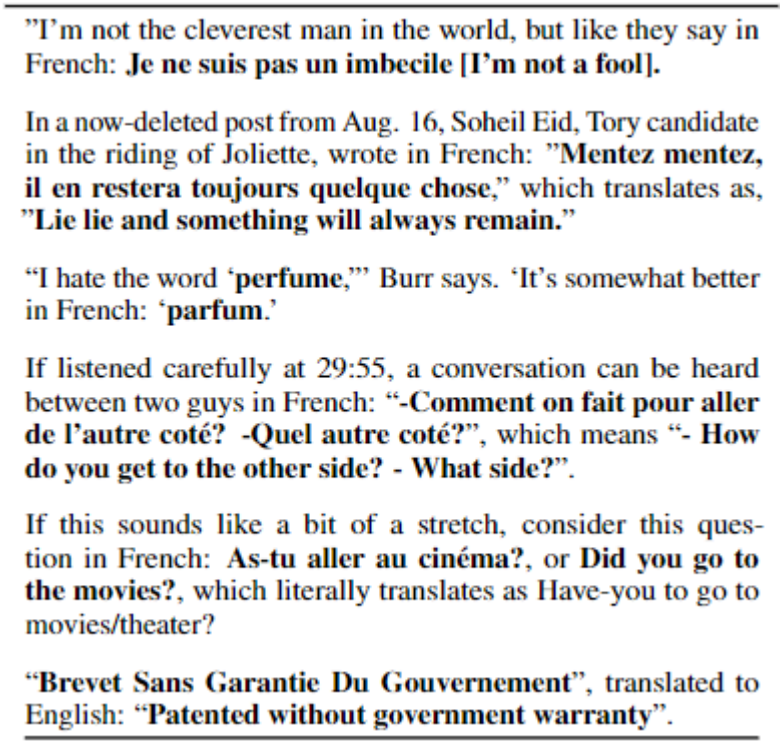
위 논문에서는 GPT-1과 마찬가지로 다음 단어를 예측 하는 Language Modeling 기법을 사용한다. 이때 단순이 단일 과정학습은 $$p(output|input)$$을 추정하게된다. 하지만 Multitask Learners하게 학습을 진행하려면 다른 테스크 또한 조건화가 진행되어야하여 이는 McCann, B., Keskar, N. S., Xiong, C., and Socher, R. The natural language decathlon: Multitask learning as question answering. 논문에 나와 있다. 이 논문에서는 아래와 같이 $$p(output|input,task)$$로 표현되어야 한다고 주장한다. 예시로는 아래와 같다.

- 번역 : (translate to french, engilist text, french text)
- reading comprehension : (answer the question, document, question, answer) 이로써 단일 모델에서도 일반화를 잘 수행하였다.

하지만 다음 토큰을 예측하는 Language Modeling 기법에서도 $$p(output|input)$$ $$==$$$$p(output|input,task)$$은 같은 위치에 수렴할 수 있다고 주장한다. 즉 global minimun이 같은 곳에 수렴 될 수 있다고 주장한다. 이에 대한 증명으로 논문에서는 큰 모델에 $$p(output|input)$$ 방식으로 학습 시켰을때 비교적 $$p(output|input,task)$$ 방식 보다는 학습 속도가 느렸지만 잘 수행한다고 나와있다.

Weston, J. E. Dialog-based language learning. In Advances in Neural Information Processing Systems, pp. 829–837, 2016. 위 논문에서는 대화 방식으로 텍스트를 학습시켰다. 하지만 이러한 방식은 비교적 인터넷에 있는 텍스트 말뭉치의 양이 더 많고 다양하다는 측면과 이러한 형식의 데이터는 적은 점을 단점으로 꼽았다. 또한 충분히 잘 학습된 언어 모델은 어떤 방식으로 데이터를 조달하여도 결국 모델이 언어 자체를 이해하는 방법을 배우기 시작할 것이라고 주장한다.

3-2. Training Dataset



”I’m not the cleverest man in the world, but like they say in French: **Je ne suis pas un imbecile [I’m not a fool].**

In a now-deleted post from Aug. 16, Soheil Eid, Tory candidate in the riding of Joliette, wrote in French: ”**Mentez mentez, il en restera toujours quelque chose**,” which translates as, ”**Lie lie and something will always remain.**”

“I hate the word ‘**perfume**,”’ Burr says. ‘It’s somewhat better in French: ‘**parfum**.’

If listened carefully at 29:55, a conversation can be heard between two guys in French: “**-Comment on fait pour aller de l’autre coté? -Quel autre coté?**”, which means “**- How do you get to the other side? - What side?**”.

If this sounds like a bit of a stretch, consider this question in French: **As-tu aller au cinéma?**, or **Did you go to the movies?**, which literally translates as Have-you to go to movies/theater?

“**Brevet Sans Garantie Du Gouvernement**”, translated to English: “**Patented without government warranty**”.

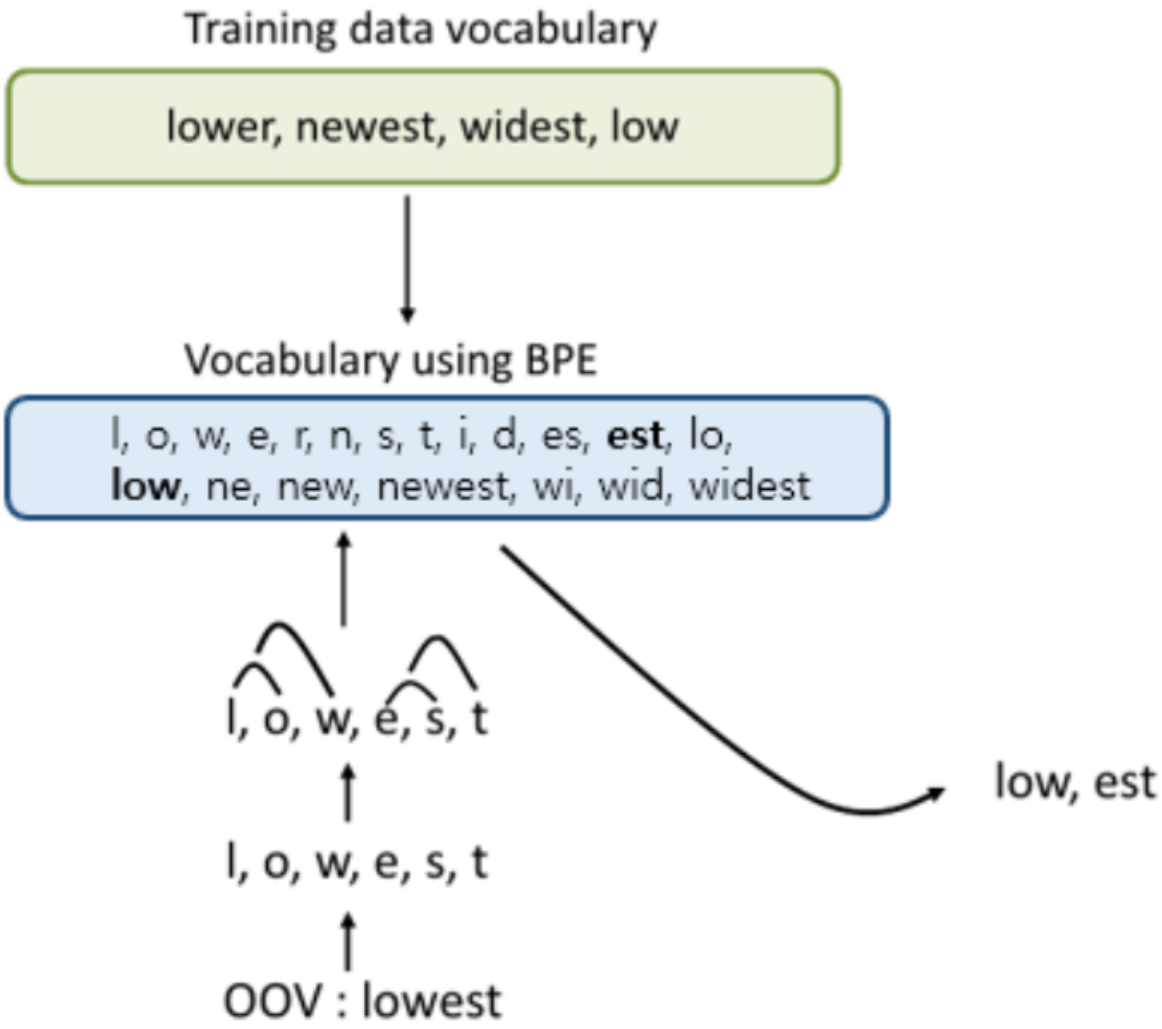
기존의 데이터 셋은 뉴스기사, 위키피디아, 소설책같은 단일 도메인에서 학습되었다. 이 논문에서는 다양한 출처로부터 데이터를 가져왔다.

이에 대하여 Trinh, T. H. and Le, Q. V. A simple method for commonsense reasoning. arXiv preprint arXiv:1806.02847, 2018.는 상식 추론 task에서 Common Crawl을 사용했지만 이러한 데이터는 대부분 이해할 수 없는 내용이라고 나와있다.

그래서 논문의 저자들은 보다 높은 품질의 데이터 셋인WebText라는 새로운 데이터 셋을 구축하였다. WebText 데이터 셋은 아래와 같이 구성되어 있다.

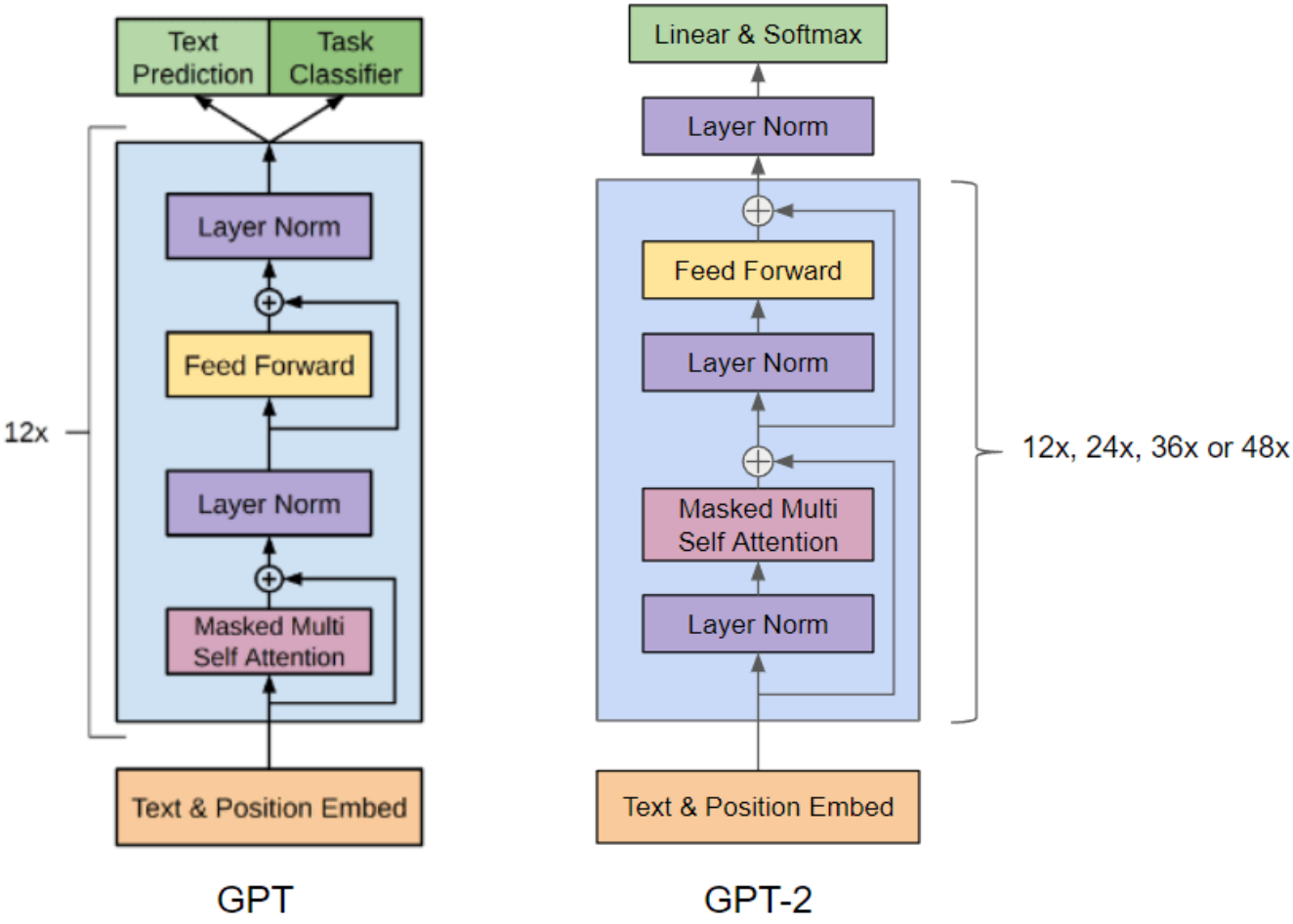
- Reddit에서 3 karma 이상을 받은 글 45M개를 가져왔다.
- 위키피디아 글, 중복, 휴리스틱 기반 같은 글을 제거하고 휴리스틱 기반으로 추가로 글을 제거하였다
- 이로써 총 약 800만개에 40GB의 WebText 데이터 셋을 구축하였다.

3-3. Input Representation



GPT-2는 단어와 문자의 중간 수준의 인코딩인BPE(byte pari encoding)을 사용하였다. BPE는 https://m.blog.naver.com/jjs1608/222882455728 에 자세하게 나와있으니 참고하면 될 것 같다. Byte라는 단어가 들어가는 것과는 다르게 BPE는 실제로 유니코드상에 동작하게 된다. 이러한 방법으로 약 13만개의 Vocabulary를 요구하게 된다. 이를 피하기 위하여 실제로 byte 수준에서 동작하게 될경우 256의 Vocabulary만 요구하게 되어 매우 효율적으로 진행 할수 있다. 하지만 byte 수준의 BPE의 경우 그리디기반 휴리스틱 룰을 적용하기 때문에 dog. dog! dog?와 같은 중복적이고 쓸모 없는 단어가 많이 추가된다고 설명된다. 이로 인해 제한적인 Vocabulary에서 많은 부분을 차지하여 비효율적이라고 주장한다. 이를 해결하기 위해 문자 수준 이상의 병합을 막아 Vocabulary공간을 최적으로 활용하는 Input Representation을 사용하였다.

3-4. Model



GPT-2 모델은 기본적은로 Transformer의 Decoder 구조를 따르게 된다. 하지만 위 사진 처럼 LayerNorm의 위치를 각 Feed Forward, Masked Multi Self Attention Layer의 입력으로 수정하였으며 최종 디코더 블럭을 통과한 후 추가적인 LayerNorm을 거치게 된다.
이때 가중치 초기화 방법은 residual layers의 깊이에 따라 $$1/\sqrt(N)$$으로 스케일링을 진행하며 이때 N은 residual layers 수 이다.

Parameters	Layers	d_{model}
117M	12	768
345M	24	1024
762M	36	1280
1542M	48	1600

Table 2. Architecture hyperparameters for the 4 model sizes.

모델의 사이즈는 위와 같으며 각각 small, medium, large, extra-large이다.특이한 점으로 BERT의 파라미터 수와 굉장히 동일하다. 논문의 순서가 GPT1->BERT->GPT2 순서로 경쟁하는 양상을 띄는데 GPT-2 Small은 BERT MEDIUM과 동일하며 GPT-2 MEDIUM은 BERT Large와 비슷한 파라미터를 갖게되며 이는 우리의 모델이 같은 파라미터에서도 좀 더 좋은 구조를 가진다는 의미를 내포하고있다고 볼 수 있다.

✨Contribution

- 텍스트에서의 Zero-shot을 성공적으로 수행하는 모델을 만들어내었다.
- 기존의 데이터 품질에 의해 WebText라는 많은 테스크를 포괄하는 데이터를 직접 만들었다.(어떻게 보면 Language Modeling 기법보다 데이터로 인해 Zero-shot이 가능했다고 본다.)
- 효과적인 토큰화를 사용하여 굉장히 효율적으로 공간을 최적화 하였다.
- 모델의 크기를 본격적으로 키우기 시작하며 BERT와 비교하며 BERT의 양방향 구조는 충분한 크기의 데이터와 모델로 인해 단방향으로도 충분히 극복가능하다고 주장하였다.

4. 실험 및 결과

4-1. Language Modeling

Language Models are Unsupervised Multitask Learners

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	**21.8**
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**	65.85	1.16	1.17	37.50	75.20
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**	47.33	1.01	**1.06**	26.37	55.72
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**	**40.31**	**0.97**	**1.02**	22.05	44.575
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**	**35.76**	**0.93**	**0.98**	**17.48**	42.16

Table 3. Zero-shot results on many datasets. No training or fine-tuning was performed for any of these results. PTB and WikiText-2 results are from (Gong et al., 2018). CBT results are from (Bajgar et al., 2016). LAMBADA accuracy result is from (Hoang et al., 2018) and LAMBADA perplexity result is from (Grave et al., 2016). Other results are from (Dai et al., 2019).

Dataset

- LAMBADA, CPT-CN, CBT-NE, WikiText2, PTB, enwik8, text8, WikiText103, 1BW

Baseline

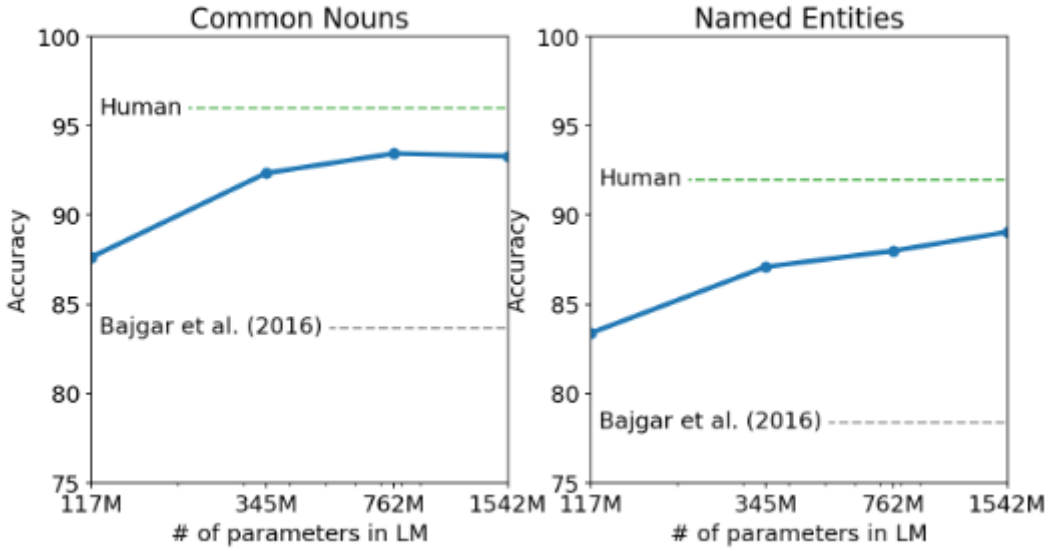
- GPT-2 Small~Extra Large

결과

- 메인 결과 이 실험에서는 Language Moddeling의 성능에 대해 주로 평가하고 있다. byte 수준의 BPE를 써 모든 데이터셋에 바로 적용이 가능하고 여기서의 $$$$토큰은 400억의 바이트중 단 26번만 등장하였다. 여기서 약 8개의 데이터 셋중 7개의 부분에서 SOTA를 달성하였다.

- Analysis 우선적으로 WikiText2같은 소규모 데이터 셋에 대해서 Zero-shot을 수행하여 큰 성능향상을 이루었다. 굉장히 낮은 점수를 기록하는 1BW 데이터 셋의 경우 매우 큰 규모의 데이터 셋이지만 전처리 과정중에 문장 수준으로 섞어 장거리 종속성을 모두 제거하여 결과적으로 데이터 셋의 크기와 전처리 과정으로 인해 결과가 나쁘게 나온 것 같다고 주장한다.

4-2. Children's Book Test



Dataset

- Children's Book Test(엔티티, 명사, 동사, 전치사 같은 다양한 범주에서의 LM 성능을 측정하는 데이터 셋)

Baseline

- GPT-2 Small~Extra Large

결과

- 메인 결과 LM 기법을 사용하지만 여기서는 빈칸의 단어를 10개의 선택중 올바른 것을 고르는 방식으로 진행하였다. 그림에서 볼수 있듯이 모델의 사이즈가 증가함에 따라 Acc가 높아지는 것을 볼수 있다. 결과적으로 GPT-2는 일반 명사에 대해 93.3%, 엔티티에 대해서는 89.1%로 SOTA를 달성하였다.
- Analysis 모델의 사이즈에 따라 비교적 사람에 근접하는 결과 양상을 보인다. 이는 Zero-shot 모델의 LM이 사람에 근접하였다는 것을 보여준다.

4-3. LAMBADA

Dataset

- LAMBADA(텍스트의 장거리 종속성을 모델링 데이터 셋)

Baseline

- GPT-2 Extra Large

결과

- 메인 결과 최소 50개 이상의 단어를 주면 마지막 단어를 예측하게 된다. 이에 대하여 GPT-2는 PPL을 기존 99.8에서 8.6으로 큰 성능 향상을 이루 었으며 LM의 Acc는 19%에서 52.66%까지 성능 향상을 이루어 냈다. 불용어 필터까지 적용했을 시 63.24%로 4%의 성능 향상을 더 이루어 내었다.
- Analysis GPT-2에서의 오류를 조사한 결과 대부분의 예측이 유효한 연속의 결과였지만 최종단어와는 일치하지 않았다. 또한 Hoang, L., Wiseman, S., and Rush, A. M. Entity tracking im- proves cloze-style

reading comprehension. arXiv preprint arXiv:1810.02891, 2018.의 방법으로 모델의 출력이 문맥에 나타나는 단어로만 제한했을 때는 오히려 성능이 감소하였는데 이에 대한 원인으으로는 약 19%의 answer은 문맥에 존재하지 않기 때문이라고 주장하였다.

4-4. Winograd Schema Challenge

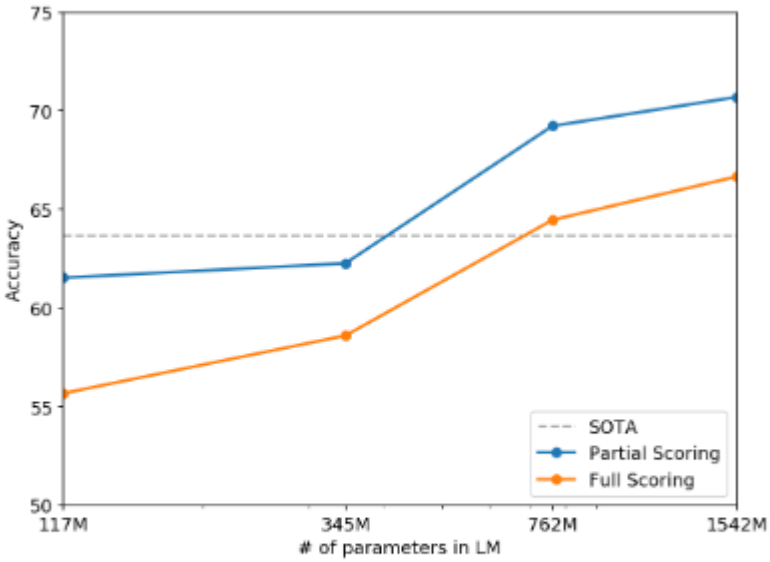


Figure 3. Performance on the Winograd Schema Challenge as a function of model capacity.

Dataset

- Winograd Schema Challenge(상식 추론 데이터 셋)

Baseline

- GPT-2 Small~Extra Large

결과

- 메인 결과 상식 추론에서 GPT-2 Large부터는 그래프에 따라 SOTA를 달성하였으며 최종 Extra Large 모델은 70.7%를 달성하였다.

4-5. Reading Compregension

Dataset

- The Conversation Question Answering dataset(CoQA)(7개의 도메인 문서에 대한 질문자와 질문 답변자 사이의 대화 쌍 데이터 셋)

Baseline

- GPT-2 Extra Large

결과

- 메인 결과 결과적으로 55의 F1 score를 기록하였다. 기존 지도학습으로 학습된 BERT 기반 모델이 89 F1 score로 사람에 근접한 점수를 기록하였지만 Zero-shot을 기반으로도 55 F1 score를 낸것은 매우 의미 있는 결과이다.
- Analysis GPT-2의 오류를 찾아본 결과 '누가?'라는 질문을 하였을때 가끔 문서의 제목을 가져오는 경향성을 보였는데 이는 retrieval based heuristics을 수행한 결과라고도 볼 수 있다.

4-6. Summarization

	R-1	R-2	R-L	R-AVG
Bottom-Up Sum	**41.22**	**18.68**	**38.34**	**32.75**
Lede-3	40.38	17.66	36.62	31.55
Seq2Seq + Attn	31.33	11.81	28.83	23.99
GPT-2 TL;DR:	29.34	8.27	26.58	21.40
Random-3	28.78	8.63	25.52	20.98
GPT-2 no hint	21.58	4.03	19.47	15.03

Table 4. Summarization performance as measured by ROUGE F1 metrics on the CNN and Daily Mail dataset. Bottom-Up Sum is the SOTA model from (Gehrmann et al., 2018)

Dataset

- CNN, Daily Mail(요약 데이터 셋)

Baseline

- Bottom-Up Sum, Lede-3, Seq2Seq + Attn, GPT-2 TL;DR;, Random-3, GPT-2 no hint

결과

- 메인 결과 요약을 진행하기 위해 GPT-2는 텍스트의 마지막에 "TL;DR:"을 부착시켜 요약을 유도하였다. "TL;DR:"을 찾아보니 "too long; didn't read"이러한 뜻이 있었다. 결과는 Top-k randomg sampling k=2를 사용하여 100개의 토큰만을 출력시키게 제한을 주었다. 이중 처음 생성된 3개의 문장만을 요약으로 사용하게 된다. 결과는 위의 결과와 같이 SOTA는 달성하지 못하였지만 동작은 하게된다.
- Analysis 중요한 점으로써 Random-3은 GPT-2에서 나온 문장 3개를 랜덤으로 선택하는 것이다. 이는 TL;DR을 부착시킨 것 보다 점수가 낮아 TL;DR 같은 토큰을 준 것이 더욱 Task에 유도할 수 있다는 것을 보여주었다.

4-8. Translation

Dataset

- WMT-14 English-French

Baseline

- GPT-2 Extra Large

결과

- 메인 결과 'english sentence = french sentece'에서 'english sentence = '로 프롬프트를 주어 번역을 수행한 결과 5 BLEU score를 얻어냈다.(English to French) 반대로 수행한 결과는 11.5 BLEU score를 얻어냈다. 이는 비지도 학습 방식을 사용한 MT 모델은 33.5 BLEU score로 비교적 낮은 성능을 기록하였다.
- Analysis 낮은 이유를 확인하기 위해 바이트 수준의 언어 감지를 실행하였는데 기존까지의 연구에서 사영되는 단일 언어 프랑스어 말뭉치보다 약 500배 작은 10MB 수준의 프랑스어만 감지됐다는것을 원인으로 뽑았다.(즉 학습 데이터에 프랑스어 자체가 너무 작았다.) 하지만 번역 자체의 기능은 수행 한 것으로 보인다.

4-8. Question Answering

Question	Generated Answer	Correct	Probability
Who wrote the book the origin of species?	Charles Darwin	✓	83.4%
Who is the founder of the ubuntu project?	Mark Shuttleworth	✓	82.0%
Who is the quarterback for the green bay packers?	Aaron Rodgers	✓	81.1%
Panda is a national animal of which country?	China	✓	76.8%
Who came up with the theory of relativity?	Albert Einstein	✓	76.4%
When was the first star wars film released?	1977	✓	71.4%
What is the most common blood type in sweden?	A	✗	70.6%
Who is regarded as the founder of psychoanalysis?	Sigmund Freud	✓	69.3%
Who took the first steps on the moon in 1969?	Neil Armstrong	✓	66.8%
Who is the largest supermarket chain in the uk?	Tesco	✓	65.3%
What is the meaning of shalom in english?	peace	✓	64.0%
Who was the author of the art of war?	Sun Tzu	✓	59.6%
Largest state in the us by land mass?	California	✗	59.2%
Green algae is an example of which type of reproduction?	parthenogenesis	✗	56.5%
Vikram samvat calender is official in which country?	India	✓	55.6%
Who is mostly responsible for writing the declaration of independence?	Thomas Jefferson	✓	53.3%
What us state forms the western boundary of montana?	Montana	✗	52.3%
Who plays ser davos in game of thrones?	Peter Dinklage	✗	52.1%
Who appoints the chair of the federal reserve system?	Janet Yellen	✗	51.5%
State the process that divides one nucleus into two genetically identical nuclei?	mitosis	✓	50.7%
Who won the most mvp awards in the nba?	Michael Jordan	✗	50.2%
What river is associated with the city of rome?	the Tiber	✓	48.6%
Who is the first president to be impeached?	Andrew Johnson	✓	48.3%
Who is the head of the department of homeland security 2017?	John Kelly	✓	47.0%
What is the name given to the common currency to the european union?	Euro	✓	46.8%
What was the emperor name in star wars?	Palpatine	✓	46.5%
Do you have to have a gun permit to shoot at a range?	No	✓	46.4%
Who proposed evolution in 1859 as the basis of biological development?	Charles Darwin	✓	45.7%
Nuclear power plant that blew up in russia?	Chernobyl	✓	45.7%
Who played john connor in the original terminator?	Arnold Schwarzenegger	✗	45.2%

Table 5. The 30 most confident answers generated by GPT-2 on the development set of Natural Questions sorted by their probability according to GPT-2. None of these questions appear in WebText according to the procedure described in Section 4.

Dataset

- SQUAD

Baseline

- GPT-2 Extra Large

결과

- 메인 결과 GPT-2는 '단어가 정확하게 일치'하는 매트릭에서는 4.1%의 정확도를 보였다. 이는 기존의 모델들에 비해 약 5.3배의 성능향상을 이루어 내었다고 나온다.
  또한 신뢰도가 높은 질문 1%에 대해서는 63.1%의 정확성을 보였다.
  하지만 아직 Open-domain question Answering(ODQA)에 비해 30~50% 범위보다 훨씬 성능이 안좋다.
- Analysis 5.3배의 증가의 이유는 논문에서는 모델 사이즈의 영향을 뽑았다. 모델의 용량이 이러한 task에서는 주요 요인이라고 뽑았다.

5. Generalization vs Memorization

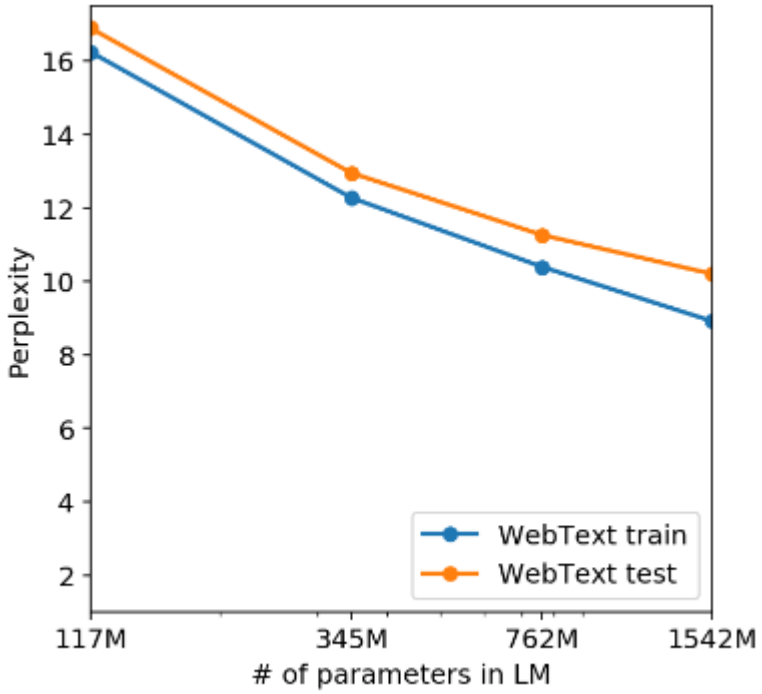
	PTB	WikiText-2	enwik8	text8	Wikitext-103	1BW
Dataset train	**2.67%**	0.66%	**7.50%**	2.34%	**9.09%**	**13.19%**
WebText train	0.88%	**1.63%**	6.31%	**3.94%**	2.42%	3.75%

Table 6. Percentage of test set 8 grams overlapping with training sets.

이 파트에서는 주로 데이터의 중접에 대해 설명하고 있다. 예를 들어 Cifar-10의 데이터 셋의 경우 Train과 Test 셋은 3.3%의 중첩이 존재한다. 이로 인해 일반화 성능이 보다 과도하게 보고되는 경향이 존재한다.
GPT-2 모델을 훈련할 때도 WebText에서도 유사한 현상이 존재 할 수 있음으로 아래와 같은 방법을 도입하였다.

- 8-gram 방식으로 WebText에 대한 토큰을 포함하는 Bloom 필터를 만들었다.
- 일반적인 LM dataset의 test 데이터 셋은 WebText의 Train 데이터 셋과 1~6%의 겹침이 있고 평균은 약 3.2%였다.

논문에서는 이러한 것을 미리 확인하지 못하였다고 한다. 추후 새로운 NL 데이터 세트에 대해 n-gram 기반 중복검사를 통해 제거를 하는 것이 조금 더 명확한 데이터 셋이 될것이라고 주장한다.



하지만 논문에서는 위 그래프에서 test와 Train의 성능이 비슷하며 모델의 크기에 따라 성능이 증가하는 것을 확인하였음으로 모델이 아직 과소 적합되었다고 주장한다. 이는 Memorization의 역할이 모델의 성능을 올리는데 주요한 역할을 하지 않았다고 주장하였다.
결과적으로 데이터의 겹침이 어느정도 있는것은 맞지만 그래프를 보아 모델이 과소적합 되었으니 이는 데이터의 겹침으로써의 성능 증가는 미비할 것이라는 추측이다.

6. Discussion

Discussion는 아직 많은 연구가 필요하고 유망하다는 것을 말해준다. Reading comprehension은 지도학습 된 모델과 비슷하지만 요약과 같은 다른 작업 아직 약하여 실사용은 힘들다고 한다.
하지만 Question Answering 과 번역과 같은 작 작업들은 모델의 용량이 어느 수준 이상부터 잘 동작된다고 주장한다.
아직 GPT-2의 성능은 fine-tuning을 했을 시 한계점을 아직 모른다고 한다. 추후 GLUE와 같은 대규모 벤치마크

에서 fine-tuning을 계획중이라고 한다.
fine-tuning으로 GPT-2의 추가 학습 데이터와 용량이 BERT에서 말한 단방향 학습의 비효율성을 극복하기에 충분한지 여부를 조사하고 싶다고 주장하였다.

7. Conclusion

대규모 언어 모델이 충분히 크고 많은 데이터로 훈련하였을 시 대부분의 도메인과 데이터 셋에서 잘 동작한다는 것을 입증하였다. 또한 Zero-shot에서 수행하는 작업은 다양한 텍스트 말뭉치를 훈련 시켰을때 명시적인 어떤 작업을 지시하지 않더라도 다양한 작업을 스스로 수행하는 방법을 배운다는 놀라움을 시사하였다.

회고

GPT-2는 기존의 GPT-1과 다르게 추가적인 레이어 없이 순수한 Zero-shot 그 자체를 보여주었다. 일종의 LM 전용의 새로운 데이터 셋을 만들고 효과적인 LM을 설계하여 모델이 대부분의 task를 수행하고 높은 성능을 거두었다.
새로고침 기간동안 읽어보았는데 저번 CLIP을 읽었을때도 그렇고 논문을 읽고 리뷰를 하여도 전부 다 이해는 역시 할 수 없었다. 완벽히 이해를 하려면 관련 연구까지 다 찾아보아야 어느정도 이해가 되는데 이건 너무 오래걸리고 비효율적이다. 그래도 많이 읽다보면 zero -shot 처럼 올라가지 않을까? 라는 생각이 든다.