

AI는 사랑은 착한 배다귀를 꿈꾸는가

네코동 고민정
2020-12-23 N3C



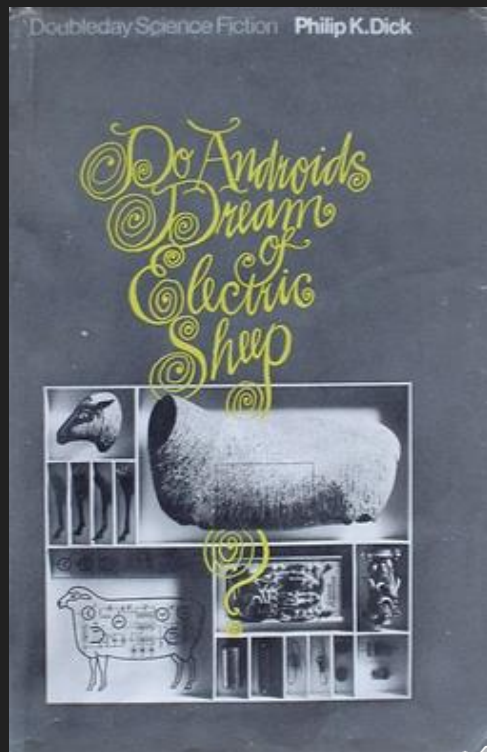
발표자 소개

- 고민정
- 넥슨, 8년차
- 네코동, 5년차
- N3C, 5년차
- 취미: 아무말
- 특기: 상상
- AI 비전문가



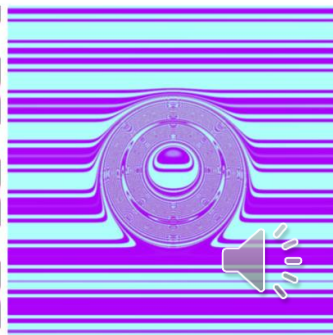
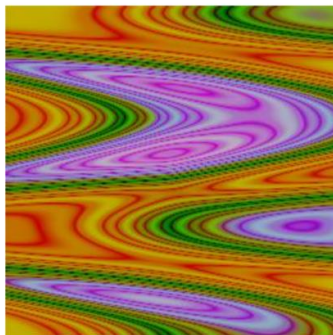
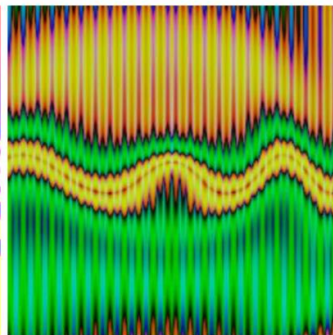
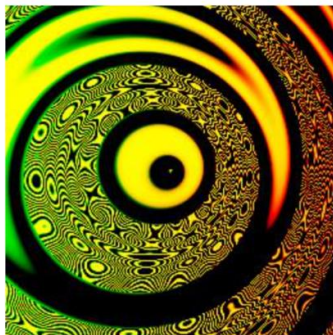
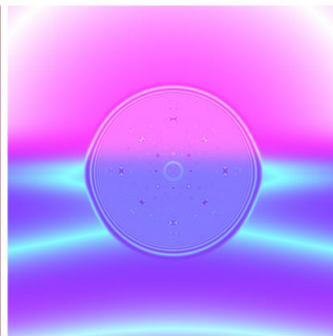
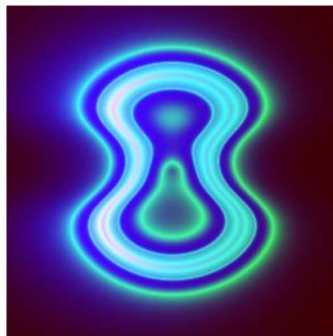
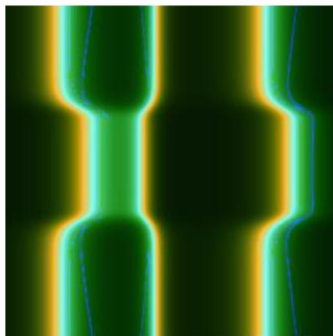
AI는 사랑은 착한 뼈다귀를 꿈꾸는가?

- 비문이 아닙니다
- <안드로이드는 전기양의 꿈꾸는가, 1968>
 - Philip K. Dick, Do Androids Dream of Electric Sheep?
 - 블레이드 러너 원작
 - 2019년 샌프란시스코 배경



AI는 꿈꾸는가

- Imagination Model



사랑은 착한 뼈다귀

- 불규칙한 단어의 조합
 - “사랑은 착한 뼈다귀”
 - “심심한 장미를 콧구멍에”
 - “파란 냄새 삼각형”

- 두억시니

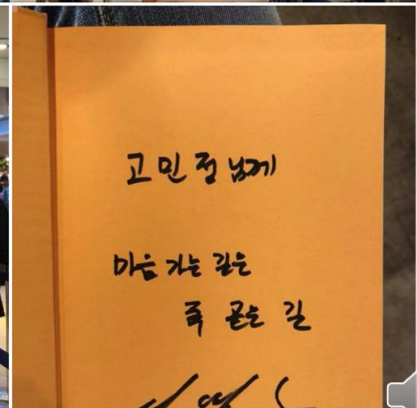
다른 종족으로부터 신을 잃은 종족으로 불린다. 모든 개체가 평균이나 기준을 발견할 수 없는 규칙적이지 못한 형태와 습관, 활동을 보이며, 심지어는 말조차도 제대로 하지 못하고 "심심한 장미를 콧구멍에"와 같이 단어를 뒤섞어서 의미 없는 말들만을 지껄인다.

- 즉, 아무말



영도빠(?)는 아닙니다만..

- 이영도
 - <드래곤라자>
 - <퓨처워커>
 - <폴라리스 랩소디>
 - <눈물을 마시는 새>
 - ...
- 딱히 자랑하려는 것은 아닙니다만..
- 싸인도 있다능..



두억시니

- <눈물을 마시는 새>
 - <피를 마시는 새>
- 신을 잃은 종족
- 신이 정해준 규칙을 잃어버려
형태도 언어도 모두 불규칙함

두억시니 (눈물을 마시는 새)

위키백과, 우리 모두의 백과사전.

두억시니는 작가 **이영도의 소설** 《**눈물을 마시는 새**》, 《**피를 마시는 새**》에서 등장하는 종족이다.

외형 [편집]

- 인간과 비슷한 신체 기관을 가졌지만 형태나 구조의 규칙이 없다.
- 다리가 있어야 할 곳에 팔이 있거나 손가락 속에 손가락이 있는 등 기괴하게 묘사된다.
- 머리가 둘 또는 셋 이상 달린 경우도 있다.
- 외형만큼 생존 규칙도 달라서 나약하게 죽기도 하고 어떤 충격을 받아도 부상을 입지 않는 경우도 있다.

특징 [편집]

- 인간의 말을 할 줄 알지만 언어 규칙도 없다. 단어의 조합에 가깝다.
- 손재주가 좋은 편이 아니고 끈기 있게 뭔가를 하지 못해서 가공된 의복을 갖추지 못하고 다닌다.
- 다른 종족의 예상보다 훨씬 적은 양의 식사를 한다.

이외 설정 [편집]

- 눈물을 마시는 새에 나오는 모종의 사건 이후 22마리만 남았다.
- 그들은 북부 대호왕의 첫번째 22금군이 된다.



두억시니 니름기계

- 아무말 머신을 만들어 보자!
 - 웬지 재미있을 것 같으니
- 랜덤
- 아무말
- 대잔치다!



창발적 랜덤

- 무한 원숭이 정리

- https://ko.wikipedia.org/wiki/%EB%AC%B4%ED%95%9C_%EC%9B%90%EC%88%AD%EC%9D%B4_%EC%A0%95%EB%A6%AC



보렐-칸텔리 보조정리의 두 번째 정리에 따르면, 충분한 시간만 주어진다면, 제멋대로 타자를 치는 **침팬지**는 필연적으로 윌리엄 셰익스피어의 **희곡** 전체를 칠 것이다.



#기술

- N모 동아리에서도 핫하던 바로 그 기술
- 한번쯤 손가락 담아보고 싶던 그 기술
- GPT-3

#기술 ★

Add a topic



45



Jul 20th

Tip: Try F to search this channel x

대규모데이터 메타학습 가능 자연어처리(NLP) 모델 **GPT-3** 이 놀랄만한 결과물들을 보여주고 있어서 요즘 뜨겁네요

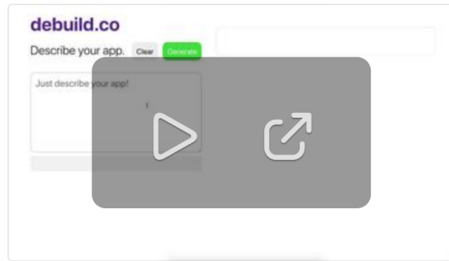
<https://twitter.com/sharifshameem/status/1284095222939451393?s=20>

Sharif Shameem @sharifshameem

I just built a **functioning** React app by describing what I wanted to GPT-3.

I'm still in awe.

Twitter | Jul 17th ▾



이건 **3달러를 더하는 버튼**, **5달러를 빼는 버튼**, 그리고 **잔고를 보여줘** 라는 설명을 적으면 결과물을 리액트앱으로 구현해주는 사례인데, 생성된 코드도 가독성 좋게 깔끔하게 나와서 놀랐네요
빠르게 프로그래머 일자리 사라지는 소리가 들리네요 🤖

👍 3 🗨️

8:04 PM

저걸 유지보수하는 직업도 생길테니, 본전일지도

👍 1 🗨️

저 조금 이상하게 생각하나요 ㅋㅋㅋㅋ?

Thursday, July 23rd ▾

7:54 AM

저 정도는 아니지만 국내 솔루션 중에 투비소프트 회사의 넥스코어? 라는 제품도 괜찮은 시도로 보입니다.
국내 SI 에서는 많이 활용하고 있다고 하네요.

Message #기술

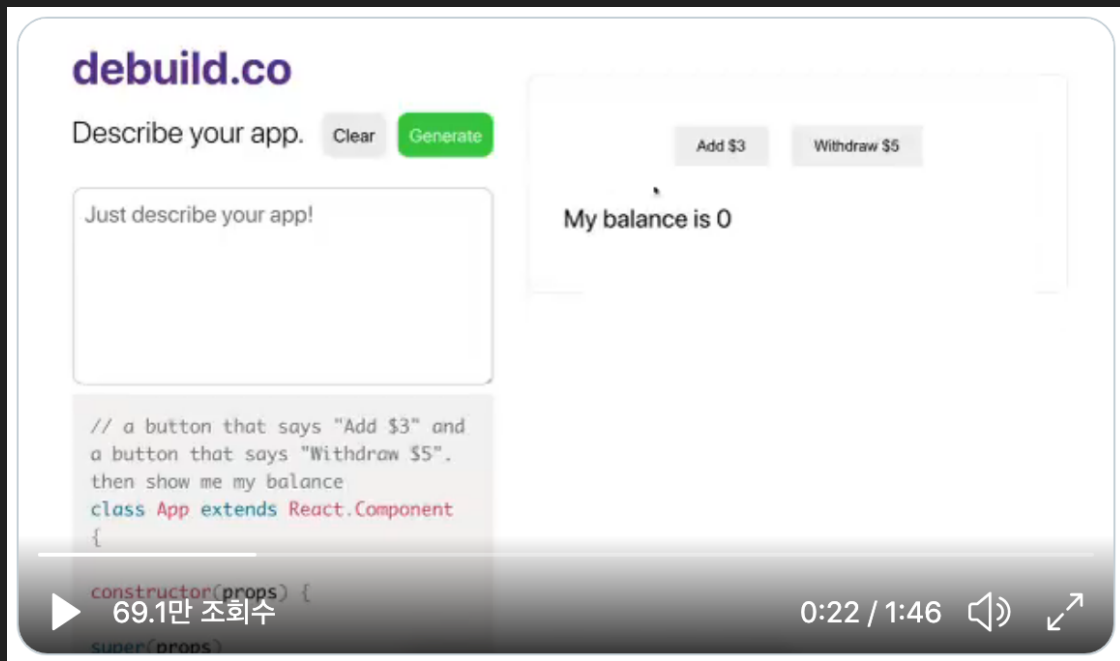
GPT-3

- OpenAI
- 강력한 자기회귀 언어모델
- 4990억 데이터 모델로 1750억 파라미터
- AI writer, AI Jukebox, AI Dungeon
- Language Models are Few-Shot Learners
 - <https://arxiv.org/abs/2005.14165>
- few-shot, one-shot, zero-shot (!)
 - ~~충분히 발달한 과학은 마법과 구분할 수 없다~~
 - 충분히 큰 모델은 파인튜닝 없이도 결과가 나올 수 있다?



Ex) 자연어 -> 코드

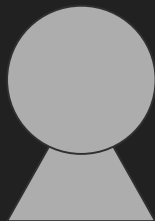
- <https://twitter.com/sharifshameem/status/1284095222939451393?s=20>



별님달님..

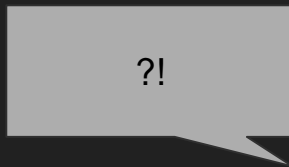


전문가의 조언



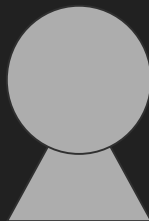
근데 GPT-3는
한국어 기반이 아니잖
아요?

두억시니 한국말 쓰는 거 아님?



?!

맞네..



- 여러분 이라서 전문가의 조언을 귀기울여야 하는 것입니다.



GPT-2

- 인공지능 기사, Fake news 우려
 - Language Models are Unsupervised Multitask Learners
 - https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf
 - 언어모델 기반
 - few-shot learning
-
- 상대적으로 데이터가 적은 두억시니 대사를 few shot 학습 시켜서 fine tuning 해보자!



KoGPT2

- 한국어 기반 GPT-2 base 모델
- SKT Conv.AI팀, Amazon Machine Learning Solutions Lab팀, GluonNLP팀
- 위키와 뉴스 등으로 2500만 이상 문장으로 학습
- 약 20GB 데이터 사용
- Github
 - <https://github.com/SKT-AI/KoGPT2>
- Demo
 - 어절을 입력하면 이후 단어 후보를 자동 생성
 - <http://52.231.69.211:8000/?text=%EB%84%A4%EC%BD%94%EB%8F%99%20%EC%97%AC%EB%9F%AC%EB%B6%84%2C%202020%EB%85%84%20%EA%B3%A0%EC%83%9D%ED%95%98%EC%85%A8%EC%8A%B5%EB%8B%88%EB%8B%A4.%20%EB%82%B4%EB%85%84%EC%97%90%EB%8A%94%20>



- <http://52020%EC920%EA9EB%8B9EB%8A9>

KoGPT-2 Explorer

B%8F%99%
8%85%84%
%8A%B5%
%97%90%

한글 어절을 입력하면 그 다음 단어를 생성해주며, 후보를 클릭함으로써 계속 생성해 낼 수 있습니다. Undo버튼을 누르면 마지막 선택이 제거됩니다. ' ' '는 공백을 의미합니다.

Sentence:

네코동 여러분, 2020년 고생하셨습니다. 내년에는

Options:

15.6% 더
7.4% 더욱
5.2% 좋은
4.9% 꼭
2.2% 다시
2.1% 여러분
2.0% 반드시
1.7% 제가
1.5% 올해보다
1.5% 정말

← Undo



준비물

- 머신러닝 학습용 장비
- 인터넷이 연결된 PC 혹은 노트북 혹은 아이패드 등
- 두억시니 대화 조각들 (파인튜닝 데이터)
- 독수리 타자 이상이 가능한 손가락
- 약간의 python 지식
- 끈기와 시간



머신러닝 학습용 장비?

— 드디어 자를 때가 온 것인가

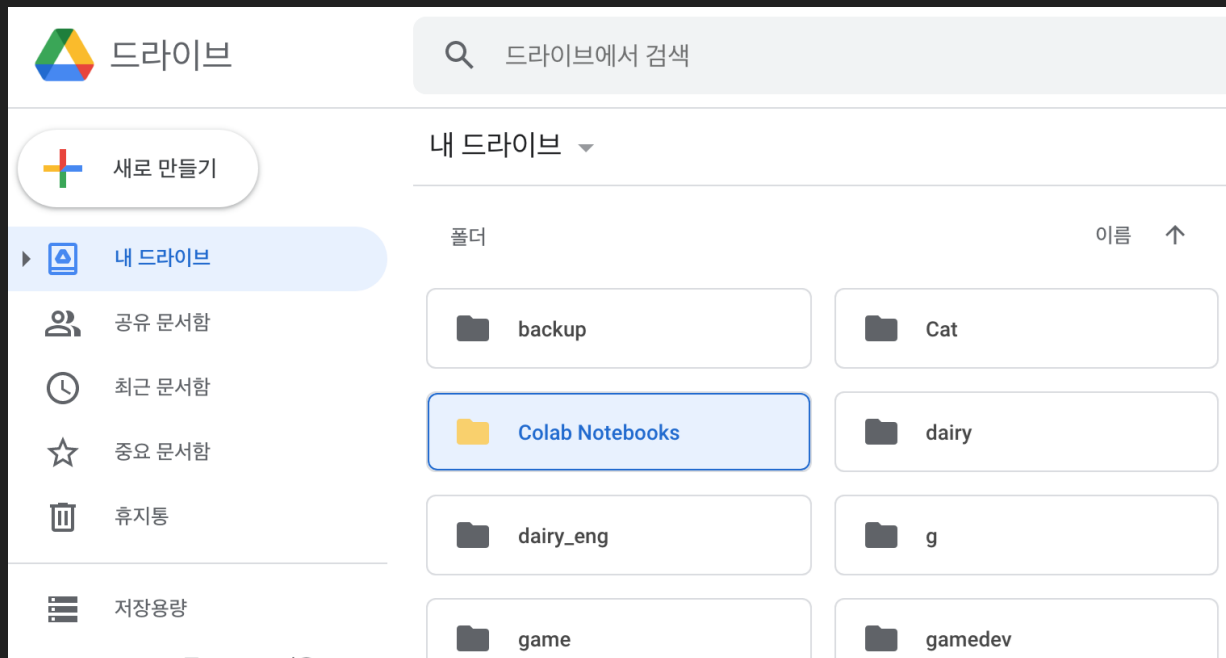


Google Colab (Google Colaboratory)

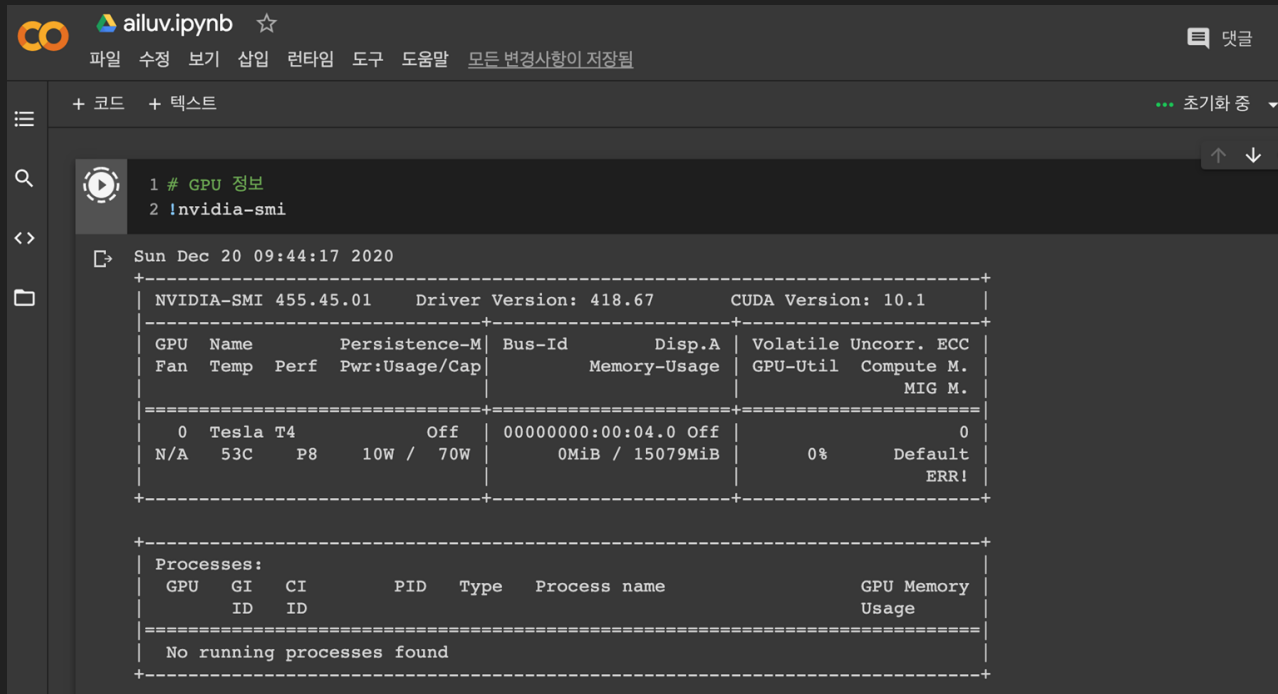
- 무료!
- Google Cloud
- Jupyter notebook
- 사양
 - GPU, TPU, CUDA
 - Tesla T4, K80
 - Intel(R) Xeon(R) CPU @ 2.20GHz / 2.30GHz
 - RAM 13GB
- 타임아웃
 - 12시간 세션 유지(연속 사용)
 - 90분 미사용시 세션 끊김
- <https://colab.research.google.com/notebooks/welcome.ipynb?hl=ko>



Google Drive 연동



Google Colab Jupyter Notebook



The screenshot shows a Google Colab Jupyter Notebook interface. The top bar includes the Colab logo, the notebook name 'ailuv.ipynb', and a star icon. Below the bar, there are tabs for '파일' (File), '수정' (Edit), '보기' (View), '삽입' (Insert), '런타임' (Runtime), '도구' (Tools), '도움말' (Help), and '모든 변경사항이 저장됨' (All changes are saved). The left sidebar contains icons for file explorer, search, expand/collapse, and a notebook icon. The main area displays a code cell with the following content:

```
1 # GPU 정보
2 !nvidia-smi
```

The output of the code cell is a terminal window showing the output of the `nvidia-smi` command. The output includes the date and time, followed by a table of GPU information.

```
Sun Dec 20 09:44:17 2020
```

NVIDIA-SMI 455.45.01 Driver Version: 418.67 CUDA Version: 10.1									
GPU Name		Persistence-M		Bus-Id	Disp.A	Volatile Uncorr. ECC		MIG M.	
Fan	Temp	Perf	Pwr:Usage/Cap		Memory-Usage	GPU-Util	Compute M.		
0	Tesla T4	Off	00000000:00:04.0 Off			0			
N/A	53C	P8	10W / 70W		0MiB / 15079MiB	0%	Default		ERR!

Below the table, there is a section for running processes:

```
Processes:
```

GPU	GI	CI	PID	Type	Process name	GPU Memory Usage
ID	ID					
No running processes found						



KoGPT2 chatbot

- GPT-2
- 공개된 한글 챗봇 데이터와 pre-trained KoGPT2를 이용한 간단한 챗봇
- 챗봇 데이터로 fine-tuning
- Simsimi
- <https://github.com/haven-jeon/KoGPT2-chatbot>
- Google Colab tutorial
- PyTorch, MXNet



Colab tutorial

```
ailuv.ipynb
파일 수정 보기 삽입 런타임 도구 도움말

+ 코드 + 텍스트
[7] Requirement already satisfied: pip&gt;=2.0.12 in /usr/local/lib/python3.6/dist-packages (from python-dateutil>=2.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.6/dist-packages (from python-dateutil>=2.0)

[8] 1 !pip install git+https://github.com/SKT-AI/KoGPT2#egg=kogpt2

Collecting kogpt2
  Cloning https://github.com/SKT-AI/KoGPT2 to /tmp/pip-install-ib4d0h0m/kogpt2
  Running command git clone -q https://github.com/SKT-AI/KoGPT2 /tmp/pip-install-ib4d0h0m/kogpt2
Building wheels for collected packages: kogpt2
  Building wheel for kogpt2 (setup.py) ... done
  Created wheel for kogpt2: filename=kogpt2-0.1.1-cp36-none-any.whl size=16045 sha256=4aee20d414240a10eef8d
  Stored in directory: /tmp/pip-ephem-wheel-cache-ksw96owg/wheels/2a/9f/62/3cba71a35387ff5da1d12e6b053b4d83
Successfully built kogpt2
Installing collected packages: kogpt2
Successfully installed kogpt2-0.1.1

1 # KoGPT2-chatbot 복사
2 # 커스텀 데이터를 사용할 두억시니 니름기계를 가져옴

[28] 1 # private 저장소 clone
2 import os
3 from getpass import getpass
4 user = getpass('GitHub user')
5 password = getpass('GitHub password')
6 # password의 #는 %23 으로 써줘야 한다 (...)
7 os.environ['GITHUB_AUTH'] = user + ':' + password
8 !git clone https://$GITHUB_AUTH@github.com/rekko/ailuv.git

GitHub user.....
GitHub password.....
Cloning into 'ailuv'...
remote: Enumerating objects: 111, done.
remote: Counting objects: 100% (111/111), done.
remote: Compressing objects: 100% (52/52), done.
remote: Total 111 (delta 61), reused 95 (delta 52), pack-reused 0
Receiving objects: 100% (111/111), 106.30 KiB | 11.81 MiB/s, done.
Resolving deltas: 100% (61/61), done.

[29] 1 # 폴더 이동
2 %cd KoGPT2-chatbot
3 %cd ailuv
```

무규칙 규칙을 학습

```
2 !CUDA_VISIBLE_DEVICES=0 python train.py --num-epoch 2 --train --batch-size 1
```

```
> using cached model  
using cached model  
using cached model  
INFO:root:contexts : 실어  
INFO:root:toked ctx: ['<usr>', '_실어', '</s>', '<unused1>', '__1', '</s>']  
INFO:root:response : 무거운 해 풀어 태어나면 개나리 웃지요!  
INFO:root:toked response : ['<s>', '_무거운', '_해', '_풀', '_어', '_태어나', '_면', '_개', '_나리', '_웃', '_지요', '!', '</s>']  
INFO:root:labels ['<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '_무거운', '_해', '_풀', '_어', '_태어나', '_면']  
INFO:root:contexts : 응  
INFO:root:toked ctx: ['<usr>', '_응', '</s>', '<unused1>', '__1', '</s>']  
INFO:root:response : 팔짝 뛰는 토끼색 칠한 재채기 세 쌍만 던져!  
INFO:root:toked response : ['<s>', '_팔', '_ 짹', '_뛰는', '_토끼', '_색', '_칠', '_한', '_재', '_채', '_기', '_세', '_쌍', '_만', '_던져', '!', '<']  
INFO:root:labels ['<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '_팔', '_짹', '_뛰는', '_토끼', '_색', '_칠']  
INFO:root:contexts : 아니오  
INFO:root:contexts : 바보  
INFO:root:toked ctx: ['<usr>', '_바보', '</s>', '<unused1>', '__2', '</s>']  
INFO:root:toked ctx: ['<usr>', '_아니', '_오', '</s>', '<unused1>', '__2', '</s>']  
INFO:root:response : 기쁠걸 한 평화! 애국자 집랑 딸꼭질!  
INFO:root:response : 일어버린 줄무늬 의표... 다율이 너무 많은 모래.  
INFO:root:toked response : ['<s>', '_기', '_쁠', '_한', '_평화', '!', '_애국', '_자', '_집', '_랑', '_딸', '_꼭', '_질', '!', '</s>']  
INFO:root:toked response : ['<s>', '_일어버린', '_줄무늬', '_의', '_표', '...', '_다', '_율', '_이', '_너무', '_많은', '_모래', '!', '</s>']  
INFO:root:labels ['<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '_기', '_쁠', '_한', '_평화', '!', '_애국']  
INFO:root:labels ['<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '_일어버린', '_줄무늬', '_의']  
INFO:root:contexts : 영  
INFO:root:toked ctx: ['<usr>', '_영', '</s>', '<unused1>', '__1', '</s>']  
INFO:root:response : 크낙새 뿌리 무침? 파란 냄새 삼각형!  
INFO:root:toked response : ['<s>', '_크', '_낙', '_새', '_뿌리', '_무', '_침', '?', '_파란', '_냄새', '_삼각', '_형', '!', '</s>']  
INFO:root:labels ['<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '<unused0>', '_크', '_낙', '_새', '_뿌리', '_무', '_침', '_?', '_파란', '_냄새', '_삼각', '_형']
```

loss loss loss

```
[Epoch 238 Batch 2/4] loss=0.0233, lr=0.0000274289, train ppl=1.024
[Epoch 250 Batch 4/4] loss=0.0432, lr=0.0000259479, train ppl=1.044
[Epoch 263 Batch 2/4] loss=0.0206, lr=0.0000244668, train ppl=1.021
[Epoch 275 Batch 4/4] loss=0.0429, lr=0.0000229858, train ppl=1.044
[Epoch 288 Batch 2/4] loss=0.0231, lr=0.0000215047, train ppl=1.023
[Epoch 300 Batch 4/4] loss=0.0425, lr=0.0000200237, train ppl=1.043
[Epoch 313 Batch 2/4] loss=0.0228, lr=0.0000185427, train ppl=1.023
[Epoch 325 Batch 4/4] loss=0.0462, lr=0.0000170616, train ppl=1.047
[Epoch 338 Batch 2/4] loss=0.0185, lr=0.0000155806, train ppl=1.019
[Epoch 350 Batch 4/4] loss=0.0447, lr=0.0000140995, train ppl=1.046
[Epoch 363 Batch 2/4] loss=0.0226, lr=0.0000126185, train ppl=1.023
[Epoch 375 Batch 4/4] loss=0.0452, lr=0.0000111374, train ppl=1.046
[Epoch 388 Batch 2/4] loss=0.0222, lr=0.0000096564, train ppl=1.022
[Epoch 400 Batch 4/4] loss=0.0471, lr=0.0000081754, train ppl=1.048
[Epoch 413 Batch 2/4] loss=0.0323, lr=0.0000066943, train ppl=1.033
[Epoch 425 Batch 4/4] loss=0.0427, lr=0.0000052133, train ppl=1.044
[Epoch 438 Batch 2/4] loss=0.0167, lr=0.0000037322, train ppl=1.017
[Epoch 450 Batch 4/4] loss=0.0437, lr=0.0000022512, train ppl=1.045
[Epoch 463 Batch 2/4] loss=0.0177, lr=0.0000007701, train ppl=1.018
[Epoch 475 Batch 4/4] loss=0.0436, lr=-0.0000007109, train ppl=1.045
[Epoch 488 Batch 2/4] loss=0.0173, lr=-0.0000021919, train ppl=1.017
[Epoch 500 Batch 4/4] loss=0.0460, lr=-0.0000036730, train ppl=1.047
```



두억시니, 사랑!



```
1 # 대화 테스트, `quit`를 입력하면 대화를 종료합니다.  
2 !CUDA_VISIBLE_DEVICES=0 python train.py --chat
```

```
... using cached model  
using cached model  
using cached model  
user > 안녕  
Simsimi > 사랑!  
user > 녀  
Simsimi > 사랑!  
user > 하하  
Simsimi > 사랑합니다.  
user > 하  
Simsimi > 사랑합니다.  
user > 머ㅇ청이  
Simsimi > 사랑!  
user > 이  
Simsimi > 사랑합니다.  
user > 바보  
Simsimi > 사랑!  
user > 보  
Simsimi > 사랑!  
user > 아니  
Simsimi > 사랑!  
user > 니  
Simsimi > 사랑!  
user > 야  
Simsimi > 사랑!  
user > 야  
Simsimi > 사랑!  
user > 사랑  
Simsimi > 사랑!  
user > 랑  
Simsimi > 사랑!
```



유해의 폭포

- 두억시니 유해
- 자아
- 자기 재생산



기대하는 바

- GPT-3

- ...



감사합니다!

