동화책 만들기

DBDBDEEP
권오현, 김다희, 오홍석, 이현진, 임수진

CONTENTS

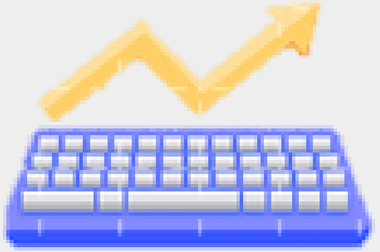
01

프로젝트 소개

01 프로젝트 소개 프로젝트 배경

스마트폰, 태블릿의 큰 인기

전자책 소비 증가



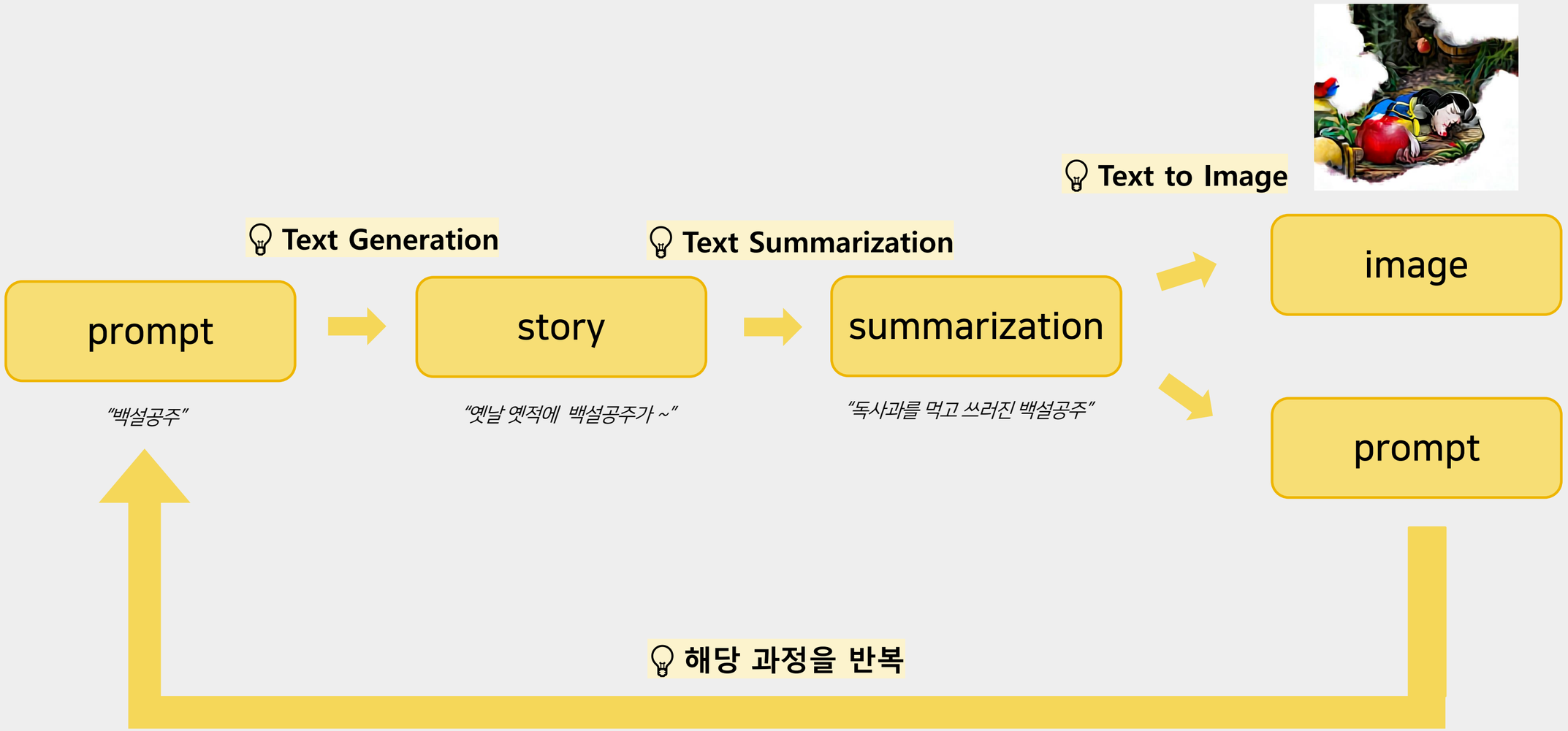
대량생산

아이에게 좋은 동화

현재의 변화된 시대상

자동화 시스템

01 프로젝트 소개 프로세스

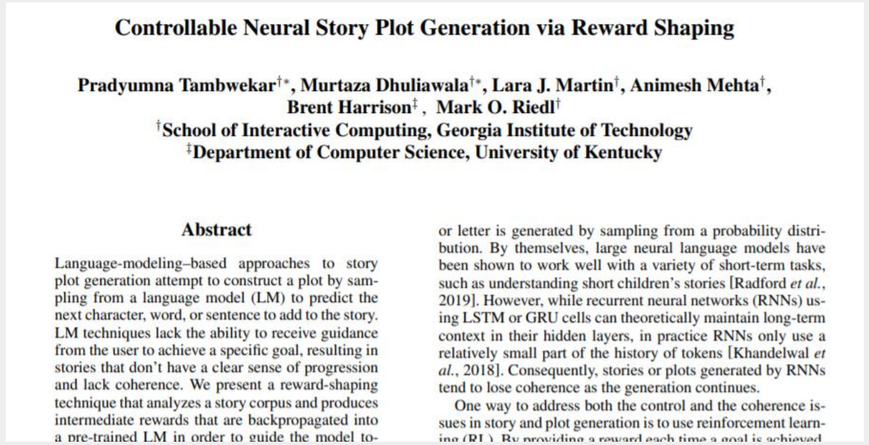


02

텍스트 생성

02 텍스트 생성 선행연구

Controllable neural story plot generation via reward shaping



Controllable Neural Story Plot Generation via Reward Shaping

Pradyumna Tambwekar[†*], Murtaza Dhuliawala[†*], Lara J. Martin[†], Animesh Mehta[†], Brent Harrison[‡], Mark O. Riedl[†]
[†]School of Interactive Computing, Georgia Institute of Technology
[‡]Department of Computer Science, University of Kentucky

Abstract

Language-modeling–based approaches to story plot generation attempt to construct a plot by sampling from a language model (LM) to predict the next character, word, or sentence to add to the story. LM techniques lack the ability to receive guidance from the user to achieve a specific goal, resulting in stories that don't have a clear sense of progression and lack coherence. We present a reward-shaping technique that analyzes a story corpus and produces intermediate rewards that are backpropagated into a pre-trained LM in order to guide the model to-

or letter is generated by sampling from a probability distribution. By themselves, large neural language models have been shown to work well with a variety of short-term tasks, such as understanding short children's stories [Radford *et al.*, 2019]. However, while recurrent neural networks (RNNs) using LSTM or GRU cells can theoretically maintain long-term context in their hidden layers, in practice RNNs only use a relatively small part of the history of tokens [Khandelwal *et al.*, 2018]. Consequently, stories or plots generated by RNNs tend to lose coherence as the generation continues.

One way to address both the control and the coherence issues in story and plot generation is to use reinforcement learn-

ing (RL). By providing a reward each time a goal is achieved

강화학습으로 통제하여 이야기 생성

보상함수 : 단어 유사도(생성된 이야기 ↔ 작가의 목표), 이야기의 응집성

단점

주제마다 새로운 학습이 필요

실제 문장이 아닌 주어, 동사, 목적어만 사용

Plan-and-Write : Towards Better Automatic Storytelling

Plan-And-Write: Towards Better Automatic Storytelling

Lili Yao[1,3*], Nanyun Peng[2*], Ralph Weischedel[2], Kevin Knight[2], Dongyan Zhao[1] and Rui Yan[1†]
liliyao@tencent.com, {npeng,weisched,knight}@isi.edu
{zhaodongyan,ruiyan}@pku.edu.cn
[1] Institute of Computer Science and Technology, Peking University
[2] Information Sciences Institute, University of Southern California, [3] Tencent AI Lab

Abstract

Automatic storytelling is challenging since it requires generating long, coherent natural language to describes a sensible sequence of events. Despite considerable efforts on automatic story generation in the past, prior work either is restricted in plot planning, or can only generate stories in a narrow domain. In this paper, we explore open-domain story generation that writes stories given a title (topic) as input. We propose a *plan-and-write* hierarchical generation framework that first plans a storyline, and then generates a story based on the storyline. We compare two planning strategies. The *dynamic* schema interweaves story planning and its surface realization in text, while the *static* schema plans out the entire storyline

Title (Given)	The Bike Accident
Storyline (Extracted)	Carrie → bike → sneak → nervous → leg
Story (Human Written)	Carrie had just learned how to ride a bike. She didn't have a bike of her own. Carrie would sneak rides on her sister's bike. She got nervous on a hill and crashed into a wall. The bike frame bent and Carrie got a deep gash on her leg.

Table 1: An example of title, storyline and story in our sys-

이야기의 줄거리를 이루는 키워드들을 바탕으로 이야기 생성

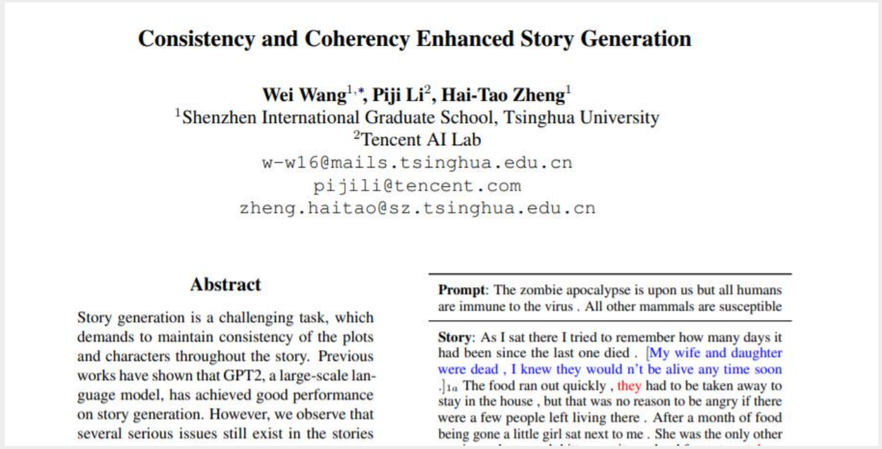
단점

단어 기반 생성으로 인한 줄거리를 단순화로 풍부한 표현X

긴 문장을 생성에 제한

02 텍스트 생성 선행연구

Consistency and Coherency Enhanced Story Generation



Consistency and Coherency Enhanced Story Generation

Wei Wang[1,*], **Piji Li**[2], **Hai-Tao Zheng**[1]
[1]Shenzhen International Graduate School, Tsinghua University
[2]Tencent AI Lab
w-w16@mails.tsinghua.edu.cn
pijili@tencent.com
zheng.haitao@sz.tsinghua.edu.cn

Abstract

Story generation is a challenging task, which demands to maintain consistency of the plots and characters throughout the story. Previous works have shown that GPT2, a large-scale language model, has achieved good performance on story generation. However, we observe that several serious issues still exist in the stories

Prompt: The zombie apocalypse is upon us but all humans are immune to the virus . All other mammals are susceptible

Story: As I sat there I tried to remember how many days it had been since the last one died . [My wife and daughter were dead , I knew they would n't be alive any time soon .]1a The food ran out quickly , they had to be taken away to stay in the house , but that was no reason to be angry if there were a few people left living there . After a month of food being gone a little girl sat next to me . She was the only other

2 step learning을 제안

이야기 생성의 Consistency와 Coherency를 향상

1단계 : Story Plot과 사건을 묘사하는 이야기의 개요를 구성

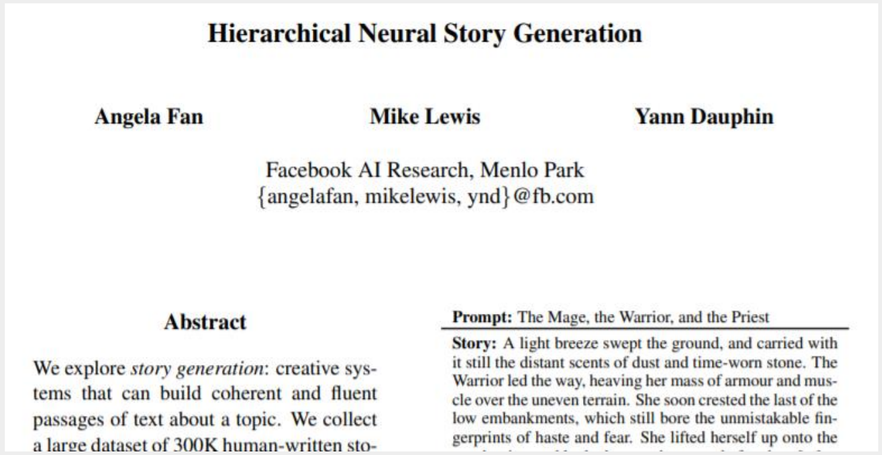
2단계 : 이를 이용하여 완전한 이야기로 확장

단점

이야기의 개요를 직접 구성해야 함

개요 학습 모델과 이야기를 생성 모델 따로 학습 => 비용 ↑

Hierarchical Neural Story Generation



Hierarchical Neural Story Generation

Angela Fan **Mike Lewis** **Yann Dauphin**

Facebook AI Research, Menlo Park
{angelafan, mikelewis, ynd}@fb.com

Abstract

We explore *story generation*: creative systems that can build coherent and fluent passages of text about a topic. We collect a large dataset of 300K human-written sto-

Prompt: The Mage, the Warrior, and the Priest

Story: A light breeze swept the ground, and carried with it still the distant scents of dust and time-worn stone. The Warrior led the way, heaving her mass of armour and muscle over the uneven terrain. She soon crested the last of the low embankments, which still bore the unmistakable fingerprints of haste and fear. She lifted herself up onto the

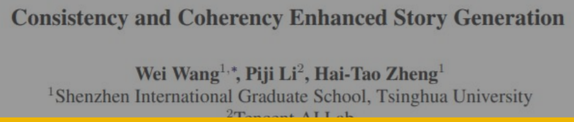
이야기의 주제(Prompt)를 생성한 후 이야기 생성

위의 논문의 경우 Conv Seq2Seq + Self Attention 모델을 제안

Reddit의 WritingPrompt 커뮤니티의 데이터로
이야기의 주제(Prompt) 생성 후에 이야기를 생성

02 텍스트 생성 선행연구

Consistency and Coherency Enhanced Story Generation



Consistency and Coherency Enhanced Story Generation

Wei Wang[1,*], **Piji Li**[2], **Hai-Tao Zheng**[1]
[1]Shenzhen International Graduate School, Tsinghua University
[2]Tencent AI Lab

2 step learning을 제안

➡ 이야기 생성의 Consistency와 Coherency를 향상

1단계 : Story Plot과 사건을 묘사하는 이야기의 개요를 구성

주제에 맞는 이야기를 생성해야 할 뿐만 아니라 생성된 이야기에 맞는 이미지도 생성해야 함

“Hierarchical Neural Story Generation” 에서 제안된 방법 사용 !

Angela Fan **Mike Lewis** **Yann Dauphin**

Facebook AI Research, Menlo Park
{angelafan, mikelewis, ynd}@fb.com

Abstract

We explore *story generation*: creative systems that can build coherent and fluent passages of text about a topic. We collect a large dataset of 300K human-written sto-

Prompt: The Mage, the Warrior, and the Priest

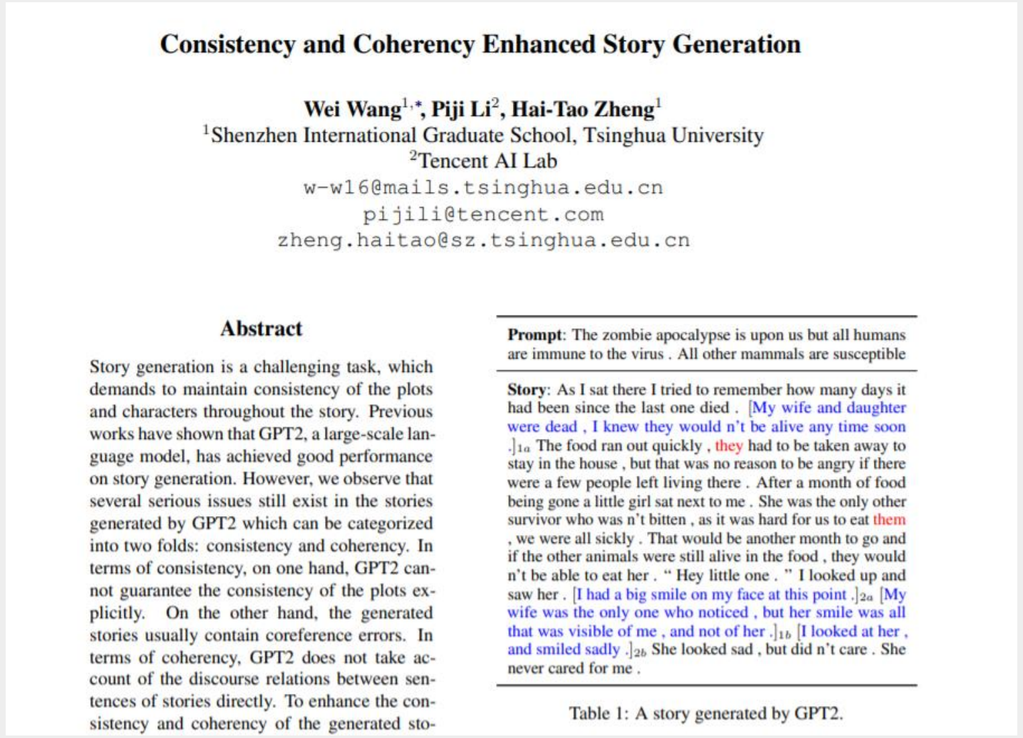
Story: A light breeze swept the ground, and carried with it still the distant scents of dust and time-worn stone. The Warrior led the way, heaving her mass of armour and muscle over the uneven terrain. She soon crested the last of the low embankments, which still bore the unmistakable fingerprints of haste and fear. She lifted herself up onto the

위의 논문의 경우 Conv Seq2Seq + Self Attention 모델을 제안

➡ Reddit의 WritingPrompt 커뮤니티의 데이터로
이야기의 주제(Prompt) 생성 후에 이야기를 생성

02 텍스트 생성 생성모델

Hierarchical Neural Story Generation



Consistency and Coherency Enhanced Story Generation

Wei Wang[1,*], Piji Li[2], Hai-Tao Zheng[1]
[1]Shenzhen International Graduate School, Tsinghua University
[2]Tencent AI Lab
w-w16@mails.tsinghua.edu.cn
pijili@tencent.com
zheng.haitao@sz.tsinghua.edu.cn

Abstract

Story generation is a challenging task, which demands to maintain consistency of the plots and characters throughout the story. Previous works have shown that GPT2, a large-scale language model, has achieved good performance on story generation. However, we observe that several serious issues still exist in the stories generated by GPT2 which can be categorized into two folds: consistency and coherency. In terms of consistency, on one hand, GPT2 cannot guarantee the consistency of the plots explicitly. On the other hand, the generated stories usually contain coreference errors. In terms of coherency, GPT2 does not take account of the discourse relations between sentences of stories directly. To enhance the consistency and coherency of the generated sto-

Prompt: The zombie apocalypse is upon us but all humans are immune to the virus . All other mammals are susceptible

Story: As I sat there I tried to remember how many days it had been since the last one died . [My wife and daughter were dead , I knew they would n't be alive any time soon .]$_{1a}$ The food ran out quickly , they had to be taken away to stay in the house , but that was no reason to be angry if there were a few people left living there . After a month of food being gone a little girl sat next to me . She was the only other survivor who was n't bitten , as it was hard for us to eat them , we were all sickly . That would be another month to go and if the other animals were still alive in the food , they would n't be able to eat her . " Hey little one . " I looked up and saw her . [I had a big smile on my face at this point .]$_{2a}$ [My wife was the only one who noticed , but her smile was all that was visible of me , and not of her .]$_{1b}$ [I looked at her , and smiled sadly .]$_{2b}$ She looked sad , but did n't care . She never cared for me .

Table 1: A story generated by GPT2.

이야기의 주제(Prompt)를 선정 후 이야기 생성

- Reddit의 WritingPrompt 커뮤니티의 데이터
  이야기의 주제(Prompt)를 생성한 후 이야기 생성
- 위의 논문의 경우 Conv Seq2Seq + Self Attention 모델을 제안

↓

Seq2Seq 문제점

Encoder-Decoder 간의 bottleneck 문제

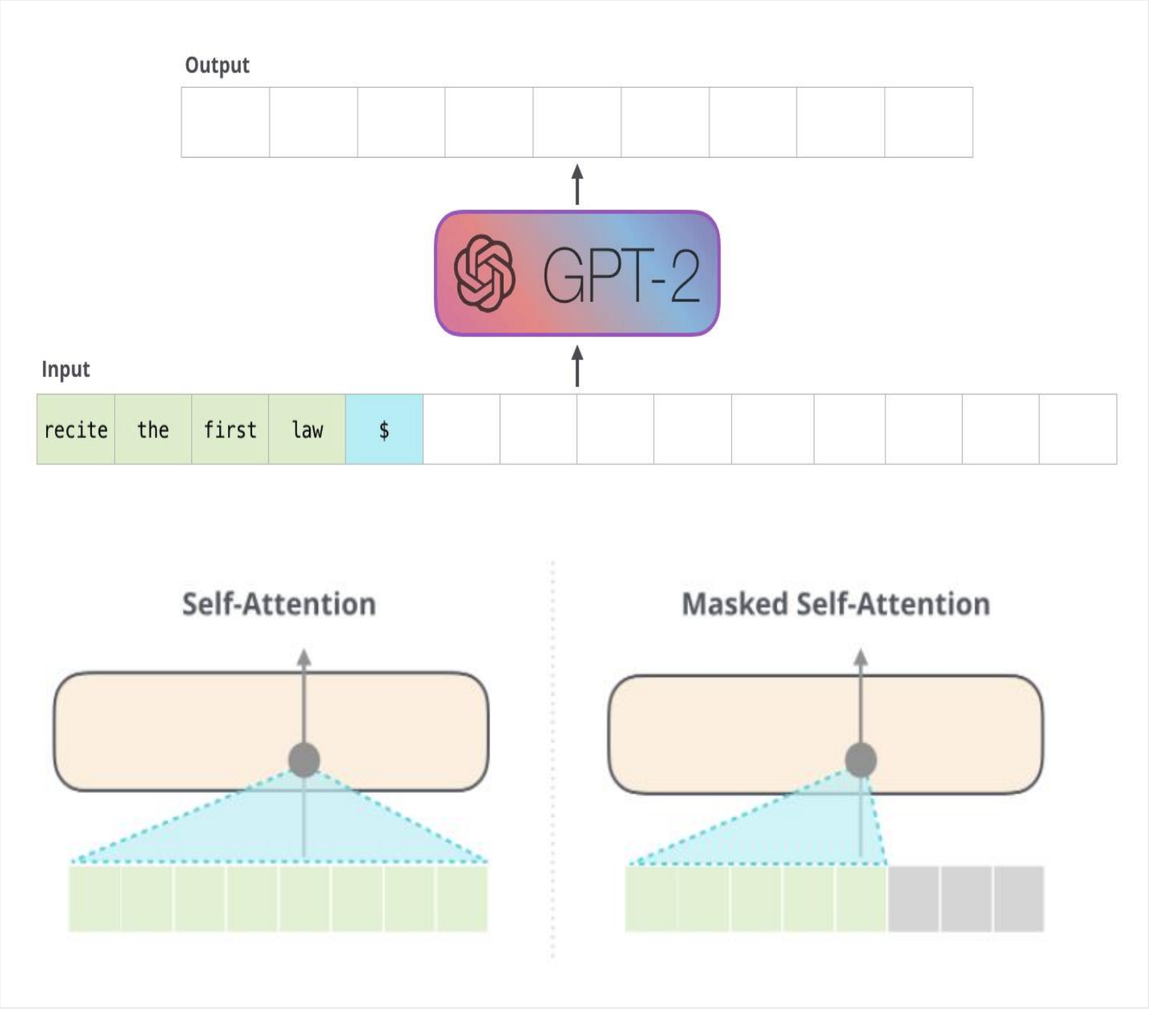
문장이 길어지면 모든 정보를 알기 어려움

Self Attention 도입에도 해결 X

Seq2Seq 구조가 아닌 Transformer의 Decoder를 사용한 GPT-2 사용

02 텍스트 생성 생성모델

GPT-2



- OpenAI에서 공개
- NLG(Natural Language Generation)에서 탁월
- Pre-trained Model

  WebText라 불리는 40GB Corpus + Crawaling Data 로 학습
- Few shot, zero shot learning에도 좋은 성능
- Fine- tuning하여 빠르게 학습 가능
- Masked Self-Attention을 사용한 Auto-Regressive Model

  -> 다음 단어의 예측 능력이 뛰어나 Generation에 강점

02 텍스트 생성 학습 데이터셋

WritingPrompt Dataset

- Reddit의 WritingPrompt 커뮤니티에서 수집된 데이터셋으로 Prompt에 해당하는 300k human-written story가 존재함

Grimm's Fairy Tales

-1812년 출간된 그림 형제의 동화집으로 209개의 이야기로 구성되어 있음 이를 활용하여 동화스럽게 생성하려고 함

위의 데이터셋은 Carnegie Mellon University에 공개 되어 있음

우리는 이 두가지 데이터셋을 이용하여
2-step learning을 진행하여 생성모델을 학습

02 텍스트 생성 전처리

1. Prompt와 Story 분리

[WP] Leonardo DiCaprio in a fit of rage begins to torpedo his own career by deliberately acting poorly and taking on bad films . H finally wins an oscar for starring in Paul Blart : Mall Cop 3 <endprompts> The wet marble floor pressed on his cheek like a thousand hands slapping his face frozen in time . Smattering piss of rain ignored his indignant mumblings . His eyes fluttered . Pin and needs ran from finger to shoulder as he pushed back against the floor , contorting his aching body into a cross legged positio . Last night was bad . He gathered that . His routine dullness of though crept inwards from the edges of his mind toward the black mist that veiled his most recent memories . He struggled to recall whatever he could n't recall but only for a moment before he

<WritingPrompt>

In olden times when wishing still helped one, there lived a king whose daughters were all beautiful, but the youngest was so beautiful that the sun itself, which has seen so much, was astonished whenever it shone in her face. Close by the king's castle lay a great dark forest, and under an old lime-tree in the forest was a well, and when the day was very warm, the king's child went out into the forest and sat down by the side of the cool fountain, and when she was bored she took a golden ball, and threw it up on high and caught it, and this ball was her favorite plaything.

Now it so happened that on one occasion the princess's golden ball did not fall into the little hand which she was holding up for it, but on to the ground beyond, and rolled straight into the water. The king's daughter followed it with her eyes, but it vanished, and the well was deep, so deep that the bottom could not be seen. At this she began to cry, and cried louder and louder, and could not be comforted. And as she thus lamented someone said to her, "What ails you, king's daughter? You weep so that even a stone would show pity."

<Grimm's tales>

WritingPrompt 데이터셋의 경우, Prompt와 Story에 대해 구성 되어 있음

→ Token을 기준으로 Prompt와 Story를 분리하여 데이터셋을 구축

Grimm's tales 데이터셋의 경우, Prompt에 대한 정보가 존재하지 않음

→ Prompt-Story 구조로 데이터셋을 구축하기 위해 문단 별로 Split을 한 후,

T5를 이용하여 문단 별로 Summarization을 생성

→ 이전 Summarization과 다음 문단을 Pair로 하여 Summarization을 통해 Story를 Generation할 수 있게 구성함

2. 특수 문자 제거

Michael did to begin with was immediately stare at the username list (or as he called it , the noob hitlist) . There they were , the usual suspects . '~` ! @ # $ % ^ & * ( ) - _ += { [ } ] |₩ : ; " ' < , > . ? /Rekter/ ? . > , < ' " ; : ₩| ] } [ { =+- _ ) ( * & ^ % $ # @ ! `~ ' , the kid who attempted to make a cool username , but ended up overusing symbols . '666 ' , the so-called 'creepypasta user ' . *Nice try , unscary skrubber . * Michael thought to himself . Then , of course , there were the unlimited FNaF , brony , Minecraft , and more fags . But there was a brand new n00b he had never seen on the hitlist before . His name was xXxHuSsEiNxXx , a very incredulous

텍스트에 존재하는 불필요한 특수문자, 웹주소, 괄호 등을 제거하고

`` 와 같은 따옴표들을 " "로 변환

02 텍스트 생성 전처리

3. GPT - 2 Tokenizer를 이용하여 Tokenizing

- Huggingface에서 제공하는 GPT2 Tokenizer를 이용하여 Tokenizing함
- 이 때 Data의 경우 긴문장을 생성하기 위해 Tokenizer의 max_length인 1024로 고정하여 Padding을 진행

→ Masked Self-Attention 특성상 Padding side를 'left'로 하여 Long Sequence가 생성될 때, <|PAD|> Token이 생성되지 않도록 함

- Pytorch의 CustomerDataset을 이용하여 시작과 끝을 알리는 Token을 추가함 <|startoftext|> Prompt <|endoftext|> Story <|endoftext|>

→ 위의 방식으로 Tokenizing할 경우 우리가 원하는 Prompt를 통해 Story를 Generation하도록 설계할 수 있음

```
<|startoftext|>the golden ball dropped to the ground near the edge of the well and rolled in. the
king\'s daughter followed it with her eyes as it sank, but the bottom could not be seen, he says.
frog stretching his thick ugly head out of water was a thorny horned \'fool\'<|endoftext|>'oh, is it
you, old waddler?" said she, "i weep because my golden ball has fallen into the well." "never mind,
do not weep," answered the frog, "i can help you but what will you give me if i fetch up your ball
again?" "whatever you like, dear frog," said she, "any of my clothes, my pearls and jewels, or even
the golden crown that i wear." "thy clothes, thy pearls and jewels, and thy golden crown are not for
me," answered the frog, "but if thou wouldst love me, and have me for thy companion and playfellow,
and let me sit by thee at table, and eat from thy plate, and drink from thy cup, and sleep in thy
little bed, if thou wouldst promise all this, then would i dive below the water and fetch thee thy
golden ball again." "oh yes," she answered, "i will promise it all, whatever you want, if you will
only get me my ball again." but she thought to herself what nonsense he talks as if he could do
anything but sit in the water and croak with the other frogs, or could possibly be any one\'s
companion.<|endoftext|>
```

02 텍스트 생성 학습

GPT - 2 Small

컴퓨팅 파워 제한으로 인해 GPT-2 Small을 이용하여 2-step learning을 진행함

Colab의 TPU와 Pytorch_xla를 통해 Parallel Learning을 진행함

* TPU의 8-core 에 batch_size = 4로 학습을 하여 총 batch_size = 32로 학습을 진행

```
import torch_xla.distributed.parallel_loader as pl
import time
import gc
import warnings
import numpy as np
warnings.filterwarnings('ignore')
WRAPPED_MODEL = xmp.MpModelWrapper(gpt2)

SERIAL_EXEC = xmp.MpSerialExecutor()
def main() :

  xm.master_print('Data Loading..... ')


  torch.manual_seed(FLAGS['seed'])
  np.random.seed(FLAGS['seed'])
  torch.backends.cudnn.deterministic = True


  def get_dataset() :
    tokenized_train = FLAGS['dataset']
    return tokenized_train

  tokenized_train = SERIAL_EXEC.run(get_dataset)

  train_sampler = torch.utils.data.distributed.DistributedSampler(
      tokenized_train,
      num_replicas=xm.xrt_world_size(),
      rank=xm.get_ordinal(),
      shuffle = True)

  # Creation of Dataloaders for testing and validation. This will be used down for training and validation stage for the model.
  train_loader = torch.utils.data.DataLoader(tokenized_train,
                                             batch_size = FLAGS['batch_size'],
                                             sampler = train_sampler,
                                             num_workers = FLAGS['num_workers'],
                                             drop_last = True)


  xm.master_print('Data Loading Clear !  ')

  xm.master_print('Model Loading .........')


  # Defining the optimizer that will be used to tune the weights of the network in the training session.
  device = xm.xla_device()
  model = WRAPPED_MODEL.to(device)
  optimizer = AdamW(params =  model.parameters(), lr=FLAGS['learning_rate'], eps=FLAGS['epsilon'])
```

```
  total_steps = len(train_loader) * FLAGS['num_epochs']
  scheduler = get_linear_schedule_with_warmup(optimizer, num_warmup_steps=FLAGS['warmup_steps'], num_training_steps=total_steps)
  xm.master_print('Model Loading Clear ! ')


  # Training loop
  def train_loop_fn(loader) :
    tracker = xm.RateTracker()
    model.train()
    total_loss = 0
    for idx,data in enumerate(loader, 0):
      optimizer.zero_grad()

      input_ids = data['input_ids'].to(device, dtype = torch.long)

      mask = data['story_attention_mask'].to(device, dtype = torch.long)

      outputs = model(input_ids = input_ids, labels = input_ids, attention_mask = mask)

      loss, logits = outputs[:2]
      total_loss += loss.item()
      loss.backward()
      if idx%20==0:
        xm.master_print(f'Epoch : {epoch}/{FLAGS["num_epochs"]} [{idx}/{len(loader)}], ₩t Loss : {loss.item(): 4f}')
      xm.optimizer_step(optimizer)
      scheduler.step(loss)
      tracker.add(FLAGS['batch_size'])
      del input_ids, mask, outputs
    xm.master_print(f'Epoch : {epoch}/{FLAGS["num_epochs"]}, ₩t Total Loss : {total_loss/len(loader): 4f}')

  train_start = time.time()
  for epoch in range(1, FLAGS['num_epochs'] + 1):
    para_loader = pl.ParallelLoader(train_loader, [device])
    train_loop_fn(para_loader.per_device_loader(device))
    if (epoch) % 200 == 0 :
      xm.rendezvous('save_model')
      xm.save(model.state_dict(), f'./fine_tuned_gpt2_generator_{epoch}_clean_tale.pth')
      xm.master_print(f'save {epoch}'s model")

  end_train_time = time.time() - train_start

  print("finished training. Train time was:", end_train_time)
```

- Optimizer - AdamW : Adam의 경우 컴퓨터 비전 task에서 일반화 성능이 떨어진다는 문제점을 해결하기 위해 제안된 Optimizer로 weight decay term을 직접적으로 추가하여 위와 같은 문제점을 해결함

- Scheduler : linear scheduler with warm up

[HyperParameters]

- Epochs : 180 (WritingPrompts – 100,000 size),
  200(Grimm's tale – 10,000 size)
- Learning_rate : 5e-4
- Epsilon : 1e-8
- Warmup_steps : 100
- Learning Time : 12hour – 18 epochs(WritingPrompts) 12hour – 200 epcohs(Grimm's tale)

→ 기존의 GPT - 2 모델의 경우 <s>text</s> 형식으로 pre-trained 했기 때문에 우리가 원하는 구조로 생성하도록 하기 위해 많은 Epochs를 통해 Fine-tuning을 함

02 텍스트 생성 학습 – WritingPrompt Dataset

```
# Generator HyperParameters
max_length = len(token_text)* 5
repetition_penalty=1.
no_repeat_ngram_size=4
min_length= len(token_text)
top_k = 40 top_p=0.95)
```

Input data :

<|startoftext|>king's daughter used to go forth into the wood and sit by the brink of the cool well. when the day was hot, she would throw it up and catch it again, and this was her favourite pastime - and it was a great night out!<|endoftext|>

<|startoftext|>king's daughter used to go forth into the wood and sit by the brink of the cool well. when the day was hot, she would throw it up and catch it again, and this was her favourite pastime - and it was a great night out!<|endoftext|> " Hey, wait a minute. You see, the world is a weird place. Sometimes you just can't keep up, you can't even take a breather. You can barely take your eyes off of the stars. You can't just sit here and stare. They're all black. " " And that's pretty much all there is to see. I mean, there's a lake, there 'd be trees all over it. It's like they just disappear into this nothingness that just ca n ' t be seen at all. It 'd be like you're in a ghost town if nothing else happened. " I sat staring at the wall in the corner of my mind. Nothing happened. It was just as it was yesterday. And I knew it was coming to an end. " Wait, why are they all black? Why didn't they just disappear? What about the stars? " I wanted to ask her this, but I had to stop myself from staring.

<Epochs – 60 Avg Perplexity - 22.73714>

<|startoftext|>king's daughter used to go forth into the wood and sit by the brink of the cool well. when the day was hot, she would throw it up and catch it again, and this was her favourite pastime - and it was a great night out!<|endoftext|> It all started when the elders decided to kill the young king and leave him alone in this cave. But what had happened? When he had last looked into his eyes, there were tears, and the fires were extinguished. For a long time the elders hid in the cavern, away from the rest of the kingdom, and they heard of an evil man, lurking far away that they could not see. But the man never came again. The king's great kingdom was ravaged, and the king decided to wait for him to return. But as he waited, the evil man took out a spear and chopped at the king's throat. The king gasped. No more could the king be summoned. No more had the king gone. And so the king began to regret his choice of action. The king wasn't evil, of course. But as evil as the evil man was, the king still craved revenge. And so revenge was an option when the evil man's mighty mightiness took hold. However, when the king had finally been resurrected

<Epochs – 120 Avg Perplexity - 8.6218>

<|startoftext|>king's daughter used to go forth into the wood and sit by the brink of the cool well. when the day was hot, she would throw it up and catch it again, and this was her favourite pastime - and it was a great night out!<|endoftext|> The clearing was quiet for several hours, save the thumping that still filled the air. After a few hours of waiting, Karl's daughter Lana took her place upon the grass. As she took her place on the edge of the clear, she thought about the past, and about how much she wished she could spend some more time with her family. She first liked the sound of the drums as they played, and the smell of paintbrush on pine needles. As she finished her sentence, she knew her mother would be coming into the clearing at the same time as she did. " Momma? " she said as she heard the soft voice of her mother begin to speak in a language she didn't understand. Karl rose from his bench, and began walking up towards her. " Mm, mam, mam! " He was shaking his head, but knew better than to say anything. His daughter walked next to her and held out a hand. She smiled, and Karl smiled back. They

<Epochs – 180 Avg Perplexity - 4.14250>

→ 학습이 진행될수록 원하는 구조의 형태로 문장이 생성되는 것을 확인하였으며 기존의 방법론들과 다르게 문장을 길게 생성할 수 있음을 보임

→ 하지만 생성된 글의 Style이 동화 같지 않기 때문에 Grimm's tale dataset을 이용하여 보다 동화스럽게 생성하려고 함

02 텍스트 생성 학습 – Grimm's Dataset

```
# Generator HyperParameters
max_length = len(token_text)* 5
repetition_penalty=1.
no_repeat_ngram_size=4
min_length= len(token_text)
top_k = 40 top_p=0.95)
```

Input data :

<|startoftext|>king's daughter used to go forth into the wood and sit by the brink of the cool well. when the day was hot, she would throw it up and catch it again, and this was her favourite pastime - and it was a great night out!<|endoftext|>

<|startoftext|>king's daughter used to go forth into the wood and sit by the brink of the cool well. when the day was hot, she would throw it up and catch it again, and this was her favourite pastime - and it was a great night out!<|endoftext|>when the day drew near, the king's daughter was to set out on her way home. she had the little oxen which she had brought with her, and carried her off with her on her back. it was as she was entering her small village, that her brothers came up recognizing her. but as they saw her disheveled and dishevelled she said to them, "thou shinest not out of the way enough yet" but the king's eyes were full of tears. the eldest, however, said to his sister, "i do not know how to be grateful enough for the money i have received from the inn, but thou must have my coat on at once, and not put it on again" but the queen would not allow her to be seen. now she was walking earnestly on a dark and lonely road, as she walked on constantly, and did not know what to do to keep in good shape. at length when she arrived at the village her brothers, who had heard the news, said "my sister is sick and

<Epochs – 40 Avg Perplexity - 8.356168>

<|startoftext|>king's daughter used to go forth into the wood and sit by the brink of the cool well. when the day was hot, she would throw it up and catch it again, and this was her favourite pastime - and it was a great night out!<|endoftext|>now it happened one day that the king's son came riding through the wood and whistled, as if he were calling his daughter to come to him, and then the king's daughter stepped out from behind the door. she had a great desire to go into the wood, and the faithful servant who had rode up to it with the three golden hairs in her hand was already standing outside it, and had to lead her into it. but the king's child stopped her and said to her, "just look at those golden hairs i have in my pocket i have still something which i want to do." the good servant had his hand on her back and wanted to take her there, but as he drew her in and took her by the hand, she drew her in also, and they sat down to the table together. for a long time the king's wife thought the old woman must be meaning some harm to her, but the faithful servant took her hand, but as she did not know what to do, he again said, "heart, do come in and be the

<Epochs – 100 Avg Perplexity - 2.32123>

<|startoftext|>king's daughter used to go forth into the wood and sit by the brink of the cool well. when the day was hot, she would throw it up and catch it again, and this was her favourite pastime - and it was a great night out!<|endoftext|> so when the sun had sunk behind the mountains some one came to the little house, who opened the door and said, "little green waitingmaid, waitingmaid with the limping leg, hop in, and carry off the basket on her back" the waitingmaid put on the little grey hanskin, and went with him to the well, and caught the golden fish in her hand. the little white dove swam quickly under the water to the shore, and carried it off with her into the little garden. here she did not come bare until, at length, she had the fish back again, carried it carefully in her great hand and set it on a perch. the king had been waiting for her for a long time, and wanted to give her a visit before his end. so he had served the king faithfully until he died, and all that he had to give the queen was his three golden apples. now it happened that the king again went out to the wood, and said to the waitinggirl, "wilt thou tell me where the

<Epochs – 200 Avg Perplexity - 1.18517>

→ 학습이 진행될수록 동화스러운 style의 이야기를 Generation하는 것을 확인

→ 기존의 Story가 아닌 새로운 내용의 Story를 생성함을 확인

03

텍스트 요약

03 텍스트 요약 T5 모델 설명

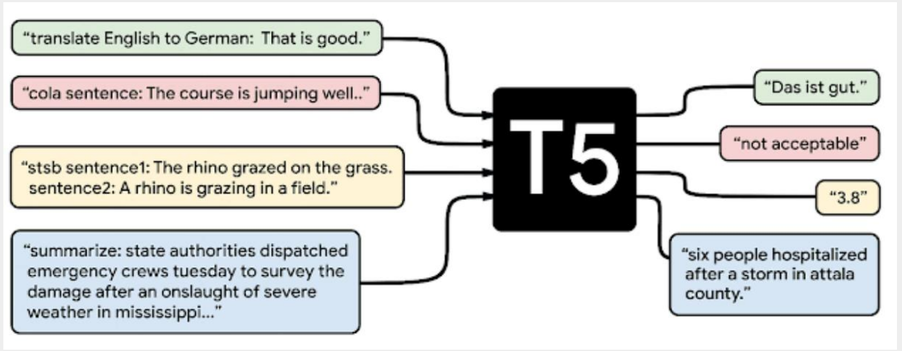
T5 (Text-To-Text Transfer Transformer)

- Colossal Clean Crawled Corpus(C4) 라는 새로운 오픈소스 사전 훈련 데이터셋 사용
- BERT 기반 모델과 달리 모든 NLP 작업을 입출력이 항상 문자열

  >“Unified Text-To-Text Transformer“
- Grimm’s tales 데이터셋의 경우 Prompt에 대한 정보가 존재X
- Prompt-Story 구조로 데이터셋을 만들기 위해 문단별로 Split

  >T5로 문단별로 Summarization 생성

  > 이전 Summarization과 다음 문단을 Pair로 하여 Summarization을 통해 Story를 Generation할 수 있게 구성



텍스트 요약 예시

a king's son was once travelling with a servant, and one of them was a handsome and tall man, and the other a short one. before the servant had to leave, and when the king's daughter was sitting at her bedside, and was about to go to bed, the king said to the tall one, "i will tell you where to pass the night. you are going to pass by my castle, and you will see the golden bird. there is one under the stairs that you can take your little flashlight and light in and just look at until you see where it is." the tall man took the torchlight from his shoulder, and set

the king's son was travelling with a servant. Before the they had to leave, he said "you are going to pass by my castle and you will see the golden bird". The tall man took the torchlight from his shoulder, and set the light on.

04

이미지 생성

04 이미지 생성 DALL-E 선정 이유

- 이미지 생성 알고리즘 소개

StackGAN

- 입력된 문장과 단어를 이용하여 가상의 이미지를 생성하며 저해상도 이미지 생성 단계와 고해상도 이미지 생성단계로 구성됨

- "StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks"
: 텍스트로부터 텍스트가 묘사하는 이미지를 실제 사진과 비슷하게 현실감있게 생성해내는데 초점

Text2Scene

- 언어로 표현한 내용에서 객체와 속성을 이끌어내어 이미지를 생성하는 모델

- 시각적인 언어 표현을 해석해 장면을 묘사함

- 만화 같은 장면부터 다양한 형식으로 합성된 이미지를 만들 수 있음

DALL-E

- 2021.02 최신모델

- Zero-Shot Learning 으로 특정 도메인이 아니더라도 우수한 이미지의 생성이 가능

* zero-shot learning
: 테스트 시간에 학습자가 훈련 중에 관찰되지 않은 클래스의 샘플을 관찰하고 그들이 속한 클래스를 예측해야 하는 기계 학습의 문제 설정

04 이미지 생성 DALL-E 선정 이유

- 이미지 생성 알고리즘 소개

StackGAN	Text2Scene	DALL-E
- 입력된 문장과 단어를 이용하여 가상의 이미지를 생성하면 정해상도	- 언어로 표현한 내용에서 객체와 속성을	- 2021.02 최신모델
사진과 비슷하게 현실감있게 생성해내는데 초점		기계 학습의 문제 설정

GAN 기반 모델은 실제 사진과 구분 못 할 정도로 정교하게 만드는 게 목적인 반면,

DALL-E는 초대규모 언어 모델인 GPT-3 기반으로 만들어져서 조금 더 창의적인 이미지를 생성할 수 있다고 판단

04 이미지 생성 DALL-E 모델 설명

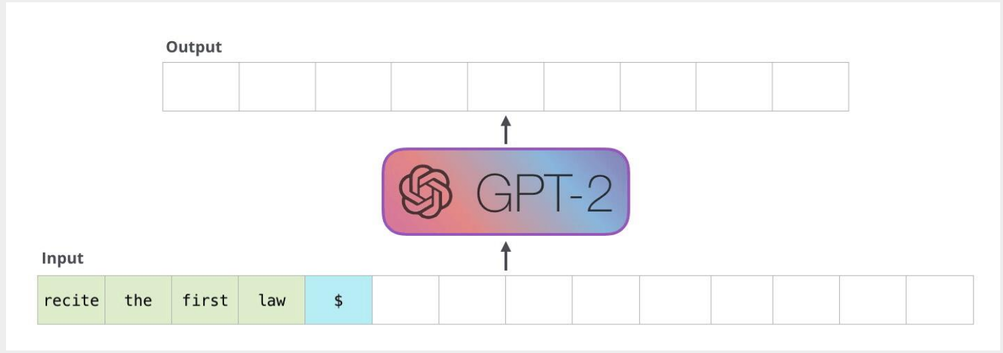
Text to Image 모델

- Text to Image 모델 (NLP와 Computer Vision이 혼합된 모델)

'아보카도 모양을 가진 의자'라는 text를 보고 생성된 image

GPT 기반 모델

- Transformer : 기존 RNN보다 Self-Attention 원리를 통해 더 빠르고 효과적으로 단어 간 관계 학습
- GPT : NLP AI로, 인간과 같은 글을 생산해내는 트랜스포머 기반 최신 언어 모델 (Text Generation)



GPT에 활용된 트랜스포머 (Decoder만 활용) → DALL-E도 같은 구조

04 이미지 생성 DALL-E 모델 설명

DALL-E의 Transformer

GPT-3의 트랜스포머와 구조가 동일하지만 인풋 데이터의 형태가 다름

→ 텍스트 토큰 + 이미지 토큰이 하나의 데이터 스트림으로 들어가는 형태

모델은 주어진 텍스트, 그리고 지금까지 예측한 픽셀값들을 전부 고려하여 다음 픽셀이 무엇일지 예측(생성)

→ 텍스트의 내용과 연관이 있으면서
이미지도 제대로 된 형태를 갖출 수 있도록
다음 픽셀을 생성할 수 있게 학습하는 것.

04 이미지 생성 DALL-E 하이퍼 파라미터 튜닝

- 목적 : ‘ 동화스러운 ‘ 이미지 생성

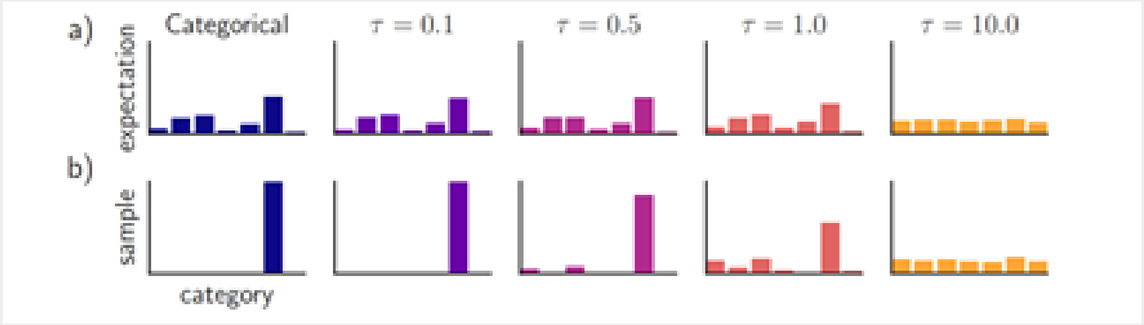
→ 적용 방법 : text input에 동화스러운 이미지 옵션을 줄 수 있는 워딩을 추가

*"As soon as Elf shakes the magic wand,
Cinderella's cloth change into a beautiful dress"*

*"A fairy tale scene that as soon as Elf shake the magic wand,
Cinderella's cloth change to a beautiful dress."*

04 이미지 생성 DALL-E 하이퍼 파라미터 튜닝

tau



tau값이 너무 높으면 무작위 sampling을 하는 역할을 하게 되는 반면,
너무 낮으면 One-hot vector처럼 categorical한 값을 갖게 됨

-> 낮은 값 (0.0) 에서 시작하여 maximum 2.0 값으로 설정하여 비교

"A Fairy tale picture of Grandmother who reads books."

tau = 1.0

tau = 2.0

tau = 1.0 이 가장 적합하다고 판단 ✔

04 이미지 생성 DALL-E 하이퍼 파라미터 튜닝

mode

- upsampling 하는 방법 조정

nearest : 주변 값을 실제 사용하는 것으로 현재 존재하는 실제 픽셀 값을 사용해야하는 경우

bilinear : 이미지와 같은 height, width의 속성을 가지는 데이터에 적합한 interpolation 방법 ✔

cutn

improve quality 관련 파라미터 → 64일때가 가장 적합하다고 판단 ✔

optimizer

eps, gamma, betas 등의 파라미터 조정 + optimizer 튜닝 → adam이 가장 적합하다고 판단 ✔

04 이미지 생성 DALL-E 하이퍼 파라미터 튜닝

- 추가 : 주어(명사)에 따른 차이 비교

"The princess fell asleep after eating a poisonous apple."

"Snow White fell asleep after eating a poisonous apple."

→ 주어가 명확한 경우 (ex. Snow White), 더 깔끔한 이미지를 생성해내는 것을 확인

05

결론

Prompt : The two kings' children

The two kings' children

A king's son was once travelling with a servant, and one of them was a handsome and tall man, and the other a short one. before the servant had to leave, and when the king's daughter was sitting at her bedside, and was about to go to bed, the king said to the tall one, "i will tell you where to pass the night. you are going to pass by my castle, and you will see the golden bird. there is one under the stairs that you can take your little flashlight and light in and just look at until you see where it is." the tall man took the torchlight from his shoulder, and set.

Prompt : the king's son was travelling with a servant. Before the they had to leave, he said "you are going to pass by my castle and you will see the golden bird". The tall man took the torchlight from his shoulder, and set the light on.

He stepped into the dark passage, and walked straight straight up to the castle. it was a great hall, and behind the beautiful golden door there sat a table in which food was still lying. on this there was a cupboard, which had wine and wine mixed with it, while on the table was a pair of red handkerchiefs which were clasped together by a golden thread. the king's son was sitting in the hall, sat in a great armchair placed on the wall, which looked as if it were made of gold, and he saw a woman statue had three golden crowns on, one for her whole body, one on each side of her, and she was so beautiful that her equal was not to be found on earth.

Prompt : the king's son was sitting in the hall, sat in a great armchair placed on the wall, which looked as if it were made of gold. the woman statue had three golden crowns on each side of her, and the statue was so beautiful.

He took the crown from her head and put it on with his left hand, and felt the gold sparkling within it. so he put it round his neck, and seated himself in the armchair. there he felt as if his heart were going to leap out of his chest, for without thinking of his fate, he placed his hand on her neck, pulled her up and kissed her. immediately a great and splendid woman moved, who was indeed beautiful but highlighted by her golden hair, with her golden eyes, and her golden nose. the statue took the king's son in her arms, carried him to her chamber, and with a violent step she brought up the golden treasure.

Prompt : statue took the king's son in her arms, carried him to chamber and brought up golden treasure to him with violent step.

A golden ring was hanging on the wall of room, and it glittered with jewels and jewels, and the jewels glittered like stars. he had not yet any idea what to do with this precious ring, and as he still did not know what to wish for it for, he felt a desire to wish so at once he put his hand into his pocket, put on the jewels in his pocket and went out with them. And when he came to the door he saw beautiful princess who wore the dress in golden.

Prompt : statue took the king's son in her arms, carried him to chamber and brought up golden treasure to him with violent step.

If the ring was to be his reward, so be it. the princess said she would be his wife, and she would give him her whole golden kingdom. so he said, "if we were to be married, and we were going to the wedding, the first one was to get his gold ring into kingdom, so the other might go with us." then the princess went out, and the wedding was solemnized with great rejoicing. he went back to the kingdom and laid the ring on his finger. the he went into the kingdom and made a great.

THE END

05 결론 및 개선방안

결론

1. 동화를 생성함에 기존에 제안된 "Hierarchical neural Story Generation" 논문 보다 좋은 성능
   - 논문 보다 **긴 Story** 생성
   - 이전 Prompt를 통해 다음 Story를 생성하므로 **통일성 있는 Story** 생성

2. Text-to-Image 모델을 적용하여 단순한 Story Generation Task가 아닌 Multi-Process Task 수행
   - 사람이 **Story를 생성하고 일러스트를 그리는데 소모되는 시간을 획기적으로 감소**시킴

개선방안

1. 우리가 제시한 방법론의 경우 Prompt를 통해 재귀적으로 Story를 생성하다 보니, Story의 통일성을 제어할 수단이 없음
   - 강화학습이나 새로운 모델을 적용하여 **End-to-End Model로의 개선** 가능성

2. 동화 데이터셋의 경우 크지 않기 때문에 다양한 Story를 생성하는데 한계
   - 좀더 큰 Corpus로 학습을 하면 풍부한 표현을 할 수 있을 것으로 보임

3. 이미지 생성의 경우 추론 시간이 10장의 이미지를 생성하는데 30분이 걸리기 때문에 이를 좀더 빠르게 생성할 수 있는 방법과 " A illustrate of " , " A fairy tale of " 와 같은 특정 워딩을 추가 하지 않아도 동화스러운 이미지를 생성할 수 있는 방법 고민
   - 실제 동화 표지 데이터셋을 통해 학습을 하면 더 자연스러운 동화책 이미지를 생성할 수 있을 것으로 보임

Thank you

DBDBDEEP 20211123