

Level 2. How to customize AI engines



<http://mindslab.ai>

1. AI 스타일링 개요

1. 데이터 정제

1. AttnGAN Train / Inference

1. AI 스타일링 실습

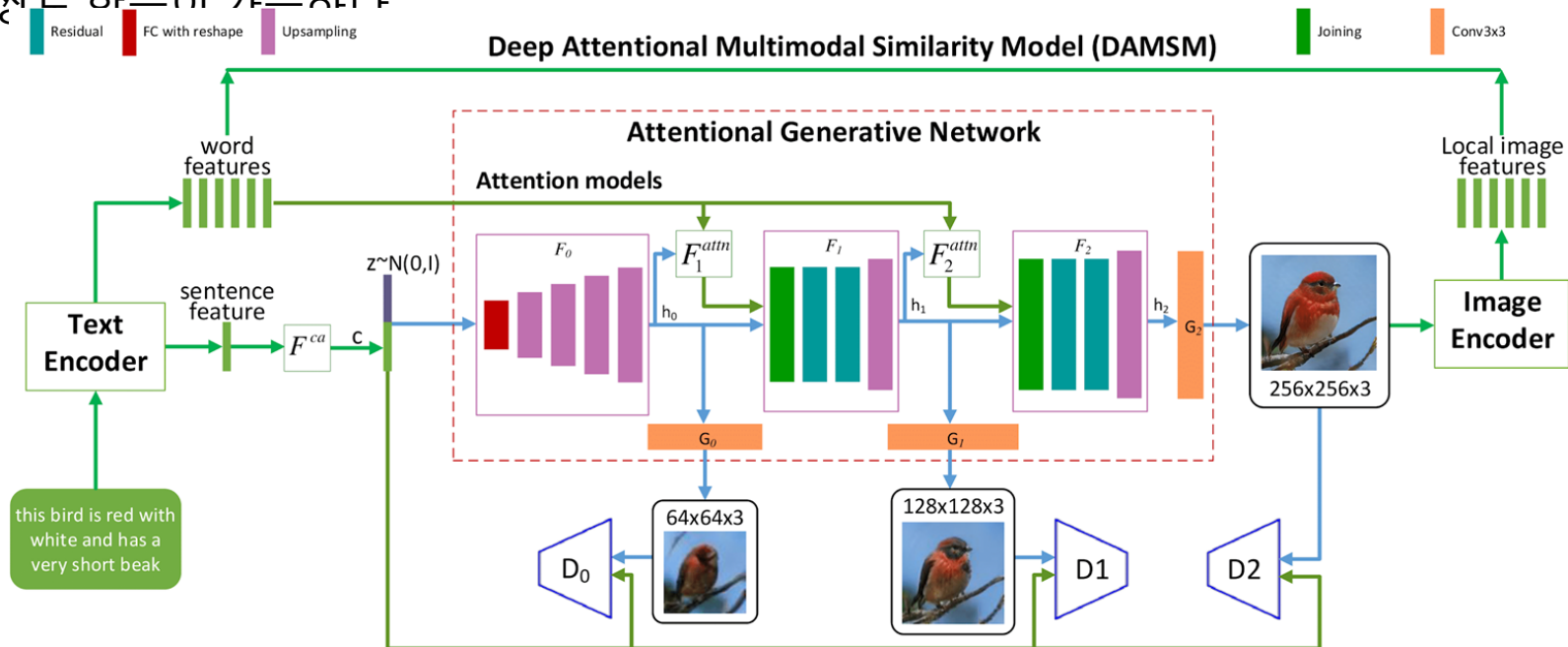


AI 스타일링이란

- 패션을 묘사하는 텍스트를 입력 받아 묘사에 맞는 옷 이미지를 생성해 내는 TTI(Text to Image) 생성형 엔진이다. 소매 길이, 옷 종류, 색, 옷감, 넥타입, 지퍼/단추 등으로 조건을 줄수 있다.

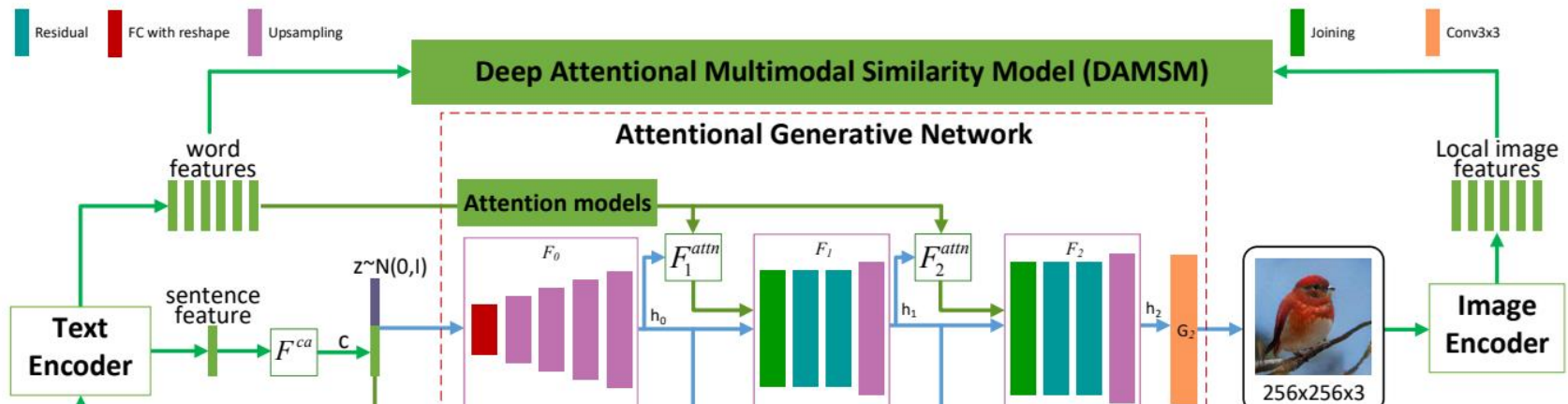
- 아키텍처는 Attention GAN으로 구성된다. 저화질의 윤곽을 먼저 생성하고, 특정 단어에 주목하며 디테일을 완성해나가는 구조이다.

- 해당 엔진을 활용해 동물 묘사-동물 이미지, 꽃 묘사-꽃 이미지 등 다른 종류의 텍스트-이미지 쌍도 학습이 가능하다



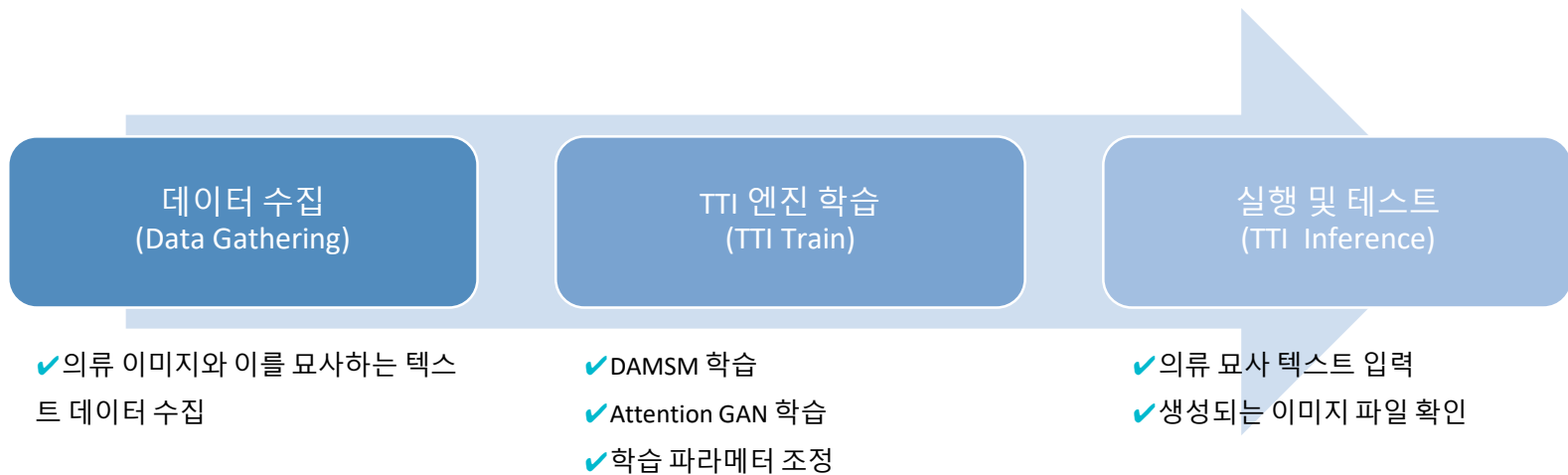
DAMSM: Deep Attentional Multimodal Similarity Model

- Text와 Image 사이의 유사도를 측정하기 위해 DAMSM이라는 아키텍처를 활용한다.
- Text와 Image를 같은 차원의 벡터로 임베딩하여 둘 사이의 유사도를 측정하는 방식이다.
- AttnGAN을 활용하고 있는 AI 스타일링 학습에서는 이 DAMSM을 먼저 사전 학습 시킨 후, 이를 활용해서 생성하는 이미지의 유사도를 측정하며 생성 모델의 학습이 진행된다



AI 스타일링 학습 및 실행 Process

- 아래의 세 단계에 걸쳐 진행된다. GAN 모델은 학습이 까다롭기 때문에 파라미터 조정에 유의해야 한다.



1. AI 스타일링 개요

1. 데이터 정제

1. AttnGAN Train / Inference

1. AI 스타일링 실습



데이터 유형

- TTI 엔진은 1) 이미지와 2) 이미지를 묘사하는 텍스트를 제공해 이를 학습하고, 이후 텍스트를 입력했을 때 이미지를 생성해내는 엔진이다.

- 예를 들어 아래와 같은 데이터에 활용할 수 있다.

- 1) 의류 사진 + 2) 패션 묘사 텍스트
- 1) 영화 포스터 + 2) 영화 소개 텍스트
- 1) 동물 사진 + 2) 동물 묘사 텍스트

Train:

묘사
텍스트



Inference:

묘사
텍스트



데이터 구축

- 이미지와 설명 텍스트를 준비한다.

1. 설명 텍스트가 존재하는 경우(영화 홍보 포스터 + 영화 설명 등)
→ 기존에 있는 데이터를 수집해서 만들 수 있다.
2. 설명 텍스트가 존재하지 않는 경우(특정 이미지)
→ 인력을 동원해 이미지를 묘사하는 캡션을 작성하고 데이터를 구축한다.

- 간단한 실험 및 연습을 위해서 공개된 데이터를 활용할 수 있다. (CUB, COCO 등).

ex) CUB 데이터 다운로드(linux)

https://drive.google.com/file/d/10_LtUP9sch09QH3s_EBAgLEctBQ5JBSJ/view (captions)
<http://www.vision.caltech.edu/visipedia/CUB-200-2011.html> (images)

ex) example caption, image (CUB 데이터)

this bird has a bright yellow body, with brown on its crown and wings.
this bird has a red breast and belly as well as a small bill.
small, roundish bird with off white breast and belly, light brown crown, brown and black colored wings.
A white bird with a black crown and yellow beak
the bird has gray crown, belly and white abdomen, with black tarsus and feet
a colorful bird with a bright yellow body, a black crown and throat, orange bill, and black primaries and secondaries.



데이터 형식

1. 이미지 파일

- 특징이 잘 나타나는 이미지를 준비한다

2. 캡션 파일

- 한 이미지 당 하나의 텍스트 파일을 만든다.
- 해당 이미지를 설명하는 문장들을 '\n'으로 구분하여 각 파일에 저장한다.

데이터 정제

- 아래에 있는 샘플을 참고하여 데이터를 준비한다.

this bird has a bright yellow body, with brown on its crown and wings.
this bird has a red breast and belly as well as a small bill.
small, roundish bird with off white breast and belly, light brown crown, brown and black colored wings.
A white bird with a black crown and yellow beak
the bird has gray crown, belly and white abdomen, with black tarsus and feet
a colorful bird with a bright yellow body, a black crown and throat, orange bill, and black primaries and secondaries.



1. AI 스타일링 개요

1. 데이터 정제

1. AttnGAN Train / Inference

1. AI 스타일링 실습



pretrain_DAMSM.py 명령 옵션

```
parser = argparse.ArgumentParser(description='Train a DAMSM network')
parser.add_argument('--cfg', dest='cfg_file',
                    help='optional config file',
                    default='cfg/DAMSM/bird.yml', type=str)
parser.add_argument('--gpu', dest='gpu_id', type=int, default=0)
parser.add_argument('--data_dir', dest='data_dir', type=str, default='')
```

- cfg: config 파일 경로. (ex: cfg/DAMSM/fashion.yml)
- gpu: 학습에 사용할 gpu id
- data_dir: 학습 데이터 위치. 위 cfg 파일에서 설정한다면 생략 가능.

DAMSM 학습 command 예시

```
python ./pretrain_DAMSM.py \  
    --cfg='cfg/DAMSM/fashion.yml' \  
    --gpu=0
```

cfg/DAMSM/.yml 파라미터 옵션

```
DATASET_NAME: 'fashions'
DATA_DIR: '../data/fashions/'

...

TRAIN:
  FLAG: True
  NET_E: '' # '../DAMSMencoders/bird/text_encoder200.pth'
  BATCH_SIZE: 48
  MAX_EPOCH: 600
  SNAPSHOT_INTERVAL: 50
  ENCODER_LR: 0.002 # 0.0002best; 0.002good; scott: 0.0007 with 0.98decay
  RNN_GRAD_CLIP: 0.25
  SMOOTH:
    GAMMA1: 4.0 # 1,2,5 good 4 best 10&100bad
    GAMMA2: 5.0
    GAMMA3: 10.0 # 10good 1&100bad

TEXT:
  EMBEDDING_DIM: 256
  CAPTIONS_PER_IMAGE: 10
```

- data_dir: 학습 데이터 위치
- train.batch_size: 학습에 사용하는 GPU 메모리가 부족하다면 조정
- train.encoder_lr: learning rate. 학습이 잘 되도록 수치 조정 필요
- text.captions_per_image: 이미지 당 캡션의 수. 준비된 데이터에 따라 조정

main.py 명령 옵션

```
parser = argparse.ArgumentParser(description='Train a AttnGAN network')
parser.add_argument('--cfg', dest='cfg_file',
                    help='optional config file',
                    default='cfg/bird_attn2.yml', type=str)
parser.add_argument('--gpu', dest='gpu_id', type=int, default=-1)
parser.add_argument('--data_dir', dest='data_dir', type=str, default='')
```

- cfg: config 파일 경로. (ex: cfg/fashion.yml)
- gpu: 학습에 사용할 gpu id
- data_dir: 학습 데이터 위치. 위 cfg 파일에서 설정한다면 생략 가능.

DAMSM 학습 command 예시 (Inference시에는 cfg파일을 바꾼 후 동일하게 실행)

```
python ./main.py \  
    --cfg='cfg/fashion.yml'  
    --gpu=0
```

cfg/bird.yml 파라미터 옵션

```
CONFIG_NAME: 'attn2'

DATASET_NAME: 'birds'
DATA_DIR: '../data/birds'
GPU_ID: 0
WORKERS: 4

TREE:
  BRANCH_NUM: 3

TRAIN:
  FLAG: True
  NET_G: '' # '../models/bird_AttnGAN2.pth'
  B_NET_D: True
  BATCH_SIZE: 20
  MAX_EPOCH: 600
  SNAPSHOT_INTERVAL: 50
  DISCRIMINATOR_LR: 0.0002
  GENERATOR_LR: 0.0002
  #
  NET_E: '../DAMSMencoders/bird/text_encoder200.pth'
  SMOOTH:
    GAMMA1: 4.0 # 1,2,5 good 4 best 10&100bad
    GAMMA2: 5.0
    GAMMA3: 10.0 # 10good 1&100bad
    LAMBDA: 5.0
```

[Train]

- train.flag: 학습시 True
- train.net_e: 사전학습 된 DAMSM encoder 경로

cfg/eval_bird.yml 파라미터 옵션

```
CONFIG_NAME: 'attn2'

DATASET_NAME: 'birds'
DATA_DIR: '../data/birds'
GPU_ID: 3
WORKERS: 1

B_VALIDATION: False # True # False
TREE:
  BRANCH_NUM: 3

TRAIN:
  FLAG: False
  NET_G: '../models/bird_AttnGAN2.pth'
  B_NET_D: False
  BATCH_SIZE: 100
  NET_E: '../DAMSMencoders/bird/text_encoder200.pth'
```

[Inference]

- train.flag: Inference시 False
- train.net_g: 학습후 생성된 모델 경로
- B_validation: batch inference 여부. True면 테스트용 캡션 전부에 대해 생성, False면 일부 생성

1. AI 스타일링 개요

1. 데이터 정제

1. AttnGAN Train / Inference

1. AI 스타일링 실습



실습 주제

- 개발환경 준비
- 데이터 셋 준비
- AI 스타일링 엔진 학습
- AI 스타일링 Inference 확인

1. 개발 환경 접속

1) 서버에 접속

- Host : 192.168.0.1
- User : tti-exercise
- Password : xxxxxx

2) XDC가 설치된 위치로 이동 (이하, '기본 위치'라 함)

```
cd /home/tti-exercise/workspace/tti
```

2. 학습 데이터 준비

1) 학습데이터를 준비하여 아래의 위치에 옮겨놓는다.

```
/home/tti-exercise/tti/...
```

```
/home/tti-exercise/tti/...
```

데이터 위치 경로

데이터 파일

2) 필요한 경우 데이터 전처리를 수행한다 ('데이터 정제' 섹션 참고)

3. DAMSM Pretrain

학습데이터 경로 등 Arguments를 정확히 설정하여 학습 명령어 실행

*실행 위치 : 기본 위치

4. XDC Train

모델 경로 등 Arguments를 정확히 설정하여 train 명령어 실행

*실행 위치 : 기본 위치

4. XDC Inference

모델 경로 등 Arguments를 정확히 설정하여 inference 명령어 실행

*실행 위치 : 기본 위치

관련 자료



- <https://github.com/davidstap/AttnGAN>
- <https://arxiv.org/abs/1711.10485>
- <http://vision.caltech.edu/visipedia/CUB-200-2011.html>

감사합니다.

MINDs Lab. AI Platform Company