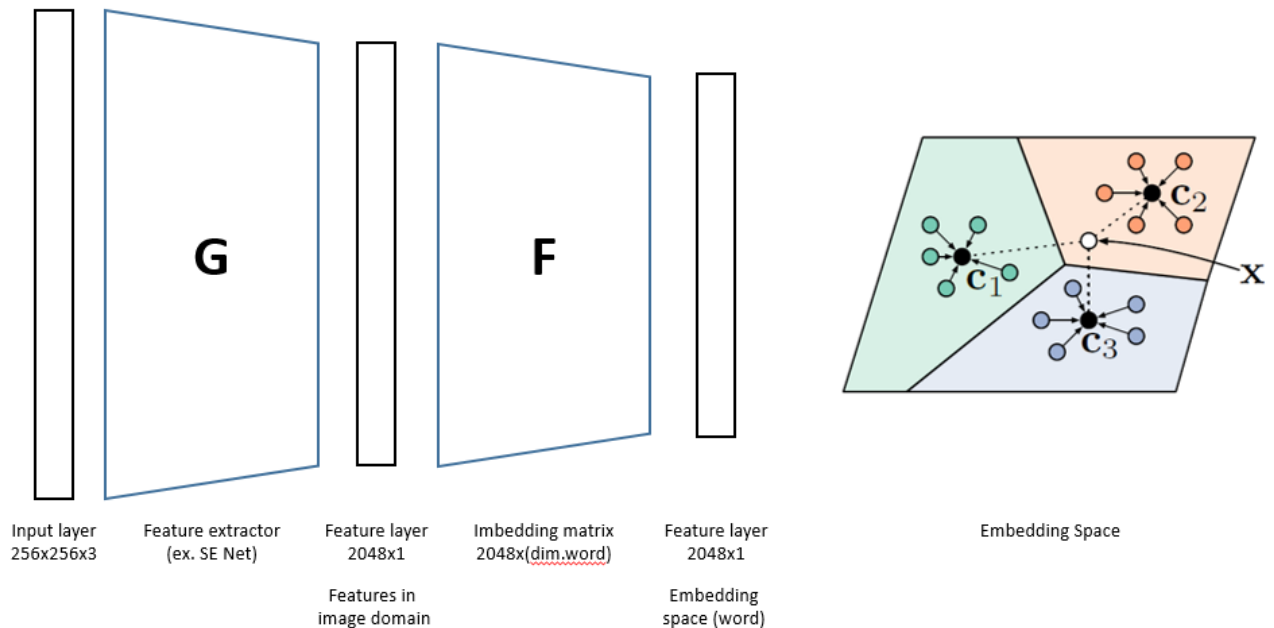
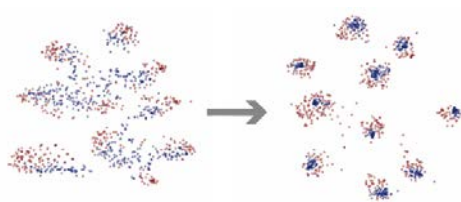


## Approach

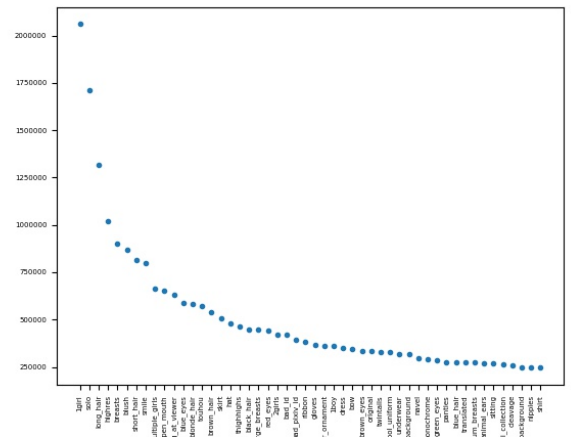
- 1 일반적인 Image with attribute 모델에서는 [1] image feature extractor  $G(x)$  뒤에 [2] image to tag mapping function  $F(x)$  를 달아서 사용한다. [1]에서 높은 레벨의 semantic feature를 추출해내는 네트워크가 많아졌고, 이를 잘 embedding 해주는  $F(x)$ 를 학습시켜야 한다.



- 2 위 방법에 대한 차선택으로,  $G(x)$  와  $F(x)$  를 합해서  $F(G(x))$ 를 학습시키는 방법도 있다. 다만 아래 2번 이슈로 인해 레이블의 질이 좋지 않기 때문에 학습에 방해요소가 될 수 있다.
  - 3 보통의 문제라면  $F(G(x))$ 에 Mapping된 포인트들이 잘 클러스터를 이루어 분포가 서로 잘 구분되도록 조정하는 연구들이 많다. 가령 <https://arxiv.org/pdf/1708.00938.pdf> 이렇게 그렇다. 그런데 우리의 문제는 레이블이 서로 포함 관계에 있는 등 서로 독립적이지 않기 때문에 오른쪽과 같은 방법을 쓰기 힘들다.
- 
- The diagram shows a cluster of points in the embedding space, with a red arrow pointing to a more refined cluster of points, indicating the process of clustering or refinement.
- 4 현재  $F(G(x))$ 는 Dual-Path Convolutional Image-Text Embedding 에서 사용된 모델이 사용되었다. 계획상  $G$ 를 학습할 수 있는 방법이 생긴다면  $G$ 를 FCN으로 구성할 계획이 있다.
  - 5 Embedding space를 Word2Vec의 도메인으로 맞추면 태그 생성에 많은 도움이 될 수 있다. 하지만 태그가 문장을 이루고 있기 때문에 word2vec을 사용해서 태그의 의미를 mapping하려면 NLP모델을 써야 한다. 마침 특정 태그들이 많이 등장하므로 그냥 태그 자체를 embedding하는 공간을 학습하는게 나을 수 있다.

## Issue

- 1 Danbooru 데이터는 별다른 클래스 분류가 없기 때문에 [1] 부분을 일반적인 방법으로 학습할 수가 없다. 바로 레이블을 예측해내는 모델을 사용할 수도 있지만 2번의 문제로 그것마저 힘들다.
  - 1.1 만화 이미지를 학습한 pretrained feature extractor를 찾고 있는 중. 아직 쓸만한 모델 찾은 것 없음.
  - 1.2 적절한 임베딩 방법을 찾으면 1.1이 필요없다. 탐색중.
- 2 Tag의 특징 관련해서 여러 이슈가 있다.
  - 2.1 [Commentary\_request], [Bad\_id] 등의 전혀 관련 없는 태그들이 존재함  
Sol) Label의 수(태그종류)가 적다면 수작업으로 위와 같은 태그들을 걸러줄 수 있다.
  - 2.2 [Hair], [Black hair] 처럼 Tag들이 서로 독립적이지 않다. 서로 포함관계에 있는 태그들이
  - 2.3 Tag가 성실하게 작성된 것이 아니므로, 태그가 없다고 해서 해당 attribute가 0이라고 보기 힘들다. 가령 [Black hair]태그를 달고 있다면 높은 확률로 흑발이겠지만, [Black hair] 태그가 없다고 해서 흑발이 아닐지는 모른다. 또한  
Sol) Negative 에 대해서 label smoothing을 적용 해주면 비교적 노이즈에 robust해진다.  
<https://arxiv.org/pdf/1512.00567.pdf> 7번항목 참고. 하지만 근본적인 해결책은 못된다.
  - 2.4 매우 많은 태그가 존재한다.  
Sol) 대다수의 태그가 200만 이미지 중 10번 이하로 나는 반면, 특정 태그들은 10만번 이상 나타난다. 이런 데이터 특성을 고려해서 Top 100 내의 태그들(10만번 이상 출현 태그 123개)에 주목한다.
- 3 어느 정도까지 성능을 낼 것인가?
  - 3.1 호랑이->고양이 하기엔 호랑이(LSTM을 포함한 NLP모델)를 만들기가 너무 힘들다.



## Related works

- About model #1 [1], [Classifying Disney Characters from Commercial Merchandise using Convolutional Neural Networks]([http://cs231n.stanford.edu/reports/2016/pdfs/265\\_Report.pdf](http://cs231n.stanford.edu/reports/2016/pdfs/265_Report.pdf))[http://cs231n.stanford.edu/reports/2016/pdfs/265\\_Report.pdf](http://cs231n.stanford.edu/reports/2016/pdfs/265_Report.pdf)

## Progress

7/16~7/22

- Data processing
  - 학습용 데이터 정제 (완료)
  - Top 100 태그 중 Irrelevant 데이터 정제 (진행중)
- Prototype 구현
  - Model (Dual path) (완료)  
<https://arxiv.org/pdf/1711.05535.pdf>
  - Data feeding & Control (완료)
- Naïve model 학습 (Base line) (진행중)
- Imbedding 방법 탐색 (진행중)