**CNN 모델링, 자연어 처리 기술을 활용한**

**신발 분류 및 태그 추천**

기간 : 20.11.18 ~ 20.11.23

**팀원 : 김성운, 김홍구,**

**윤채영, 조종운**

목차

1. 프로젝트 개요 구상
2. 이미지 데이터 - 수집, 정제, 모델링
3. 텍스트 데이터 - 수집, 정제
4. 결과 확인 및 모델 수정
5. 최종 결과 도출
6. 프로젝트 개요 구상

* 프로젝트 주제 :

CNN 모델, 자연어 처리 기술을 활용한 신발 분류 및 태그 추천

* 프로젝트 목적 :

신발 이미지를 넣었을 때 해당되는 카테고리(‘브랜드-종류’)를 예측해주고, 그에 어울리는 태그를 추천해준다.

* 카테고리 지정(‘브랜드-종류’) :

Other-Sneakers, 나이키-Running, 반스-Canvas, 슈펜-Canvas, 아디다스-Running, 아식스-Running, 엑셀시오르-Canvas, 컨버스-Canvas, 프로스펙스-Running, 휠라-Running

* + 카테고리 선정 기준 : 각 ‘브랜드-종류’ 카테고리 별로 신발의 종류가 100개가 넘는 카테고리만 사용
* 사용할 데이터 :
* 이미지 데이터 : 무신사스토어의 스니커즈&신발 카테고리 -> 우리가 지정한 카테고리에 맞는 신발 이미지 -> 카테고리별 크롤링
* 텍스트 데이터 : 무신사스토어의 스니커즈&신발 카테고리 -> 우리가 지정한 카테고리에 맞는 신발 -> 상품평순 -> 일반후기 -> 카테고리별 크롤링
* 사용 기술 :
* 파이썬 웹 크롤링
* CNN 모델링
* 자연어 처리(리뷰에서 추천 태그를 뽑아낼 목적이라 모델링은 하지 않았습니다.)

기타 DB, 웹 서버는 데이터의 용량이 크지 않아 활용하지 않았다.

* 프로젝트 진행 방식 :
* 구글 공유 폴더 -> 한 폴더내에서 데이터와 작업들을 팀원 모두가 볼 수 있게 공유했다.
* 엑셀 파일 -> 각각 단계의 진행 현황을 기록했고, 거기서 발생한 문제점 및 의문사항도 적어서 나중에 수정하기 편하게 하였다. 또한, 팀원의 의견 칸도 만들어서 그 작업을 진행하지 않는 다른 팀원의 의견도 반영될 수 있게 하였다.

2) 이미지 데이터 수집, 정제, 모델링

* 데이터 수집 출처

무신사스토어 -> 스니커즈 & 신발 카테고리 -> 우리가 지정한 카테고리에 맞는 카테고리 설정 -> 신발 이미지 전부 크롤링

* 데이터 수집 과정에서 발생한 문제점
  + ‘반스-Canvas’ 카테고리의 신발이 다른 카테고리에 비해 너무 많다.
  + 다리 나온 사진, 영수증 사진, 신발의 일부만 나온 사진 등 온전한 신발이라고 학습시키기 힘든 사진이 많다.
* 해결 방안
  + 데이터의 개수가 적은 카테고리의 이미지 -> Generator를 이용해서 Image Augmentation 기법 활용 -> 모델에 과적합될 염려가 있고, 실제로도 성능이 안 좋게 나왔다. -> Generator 사용하지 않겠다.
  + ##########사진#############
  + 정규화방법 이용 -> Min-Max Scaler 방법 이용 -> 성능이 좋아졌다. -> 이 방법을 사용하겠다.
* 분석 과정
  + 데이터 탐색 및 전처리 과정

사진 용량의 감소를 위해 500x500 -> 64x64 사이즈 조절

범주형 자료인 카테고리 -> 원 핫 인코딩을 통한 카테고리 라벨링

Min-Max Scaler를 이용한 정규화 -> 원본 이미지의 픽셀값 0~255 -> 0~1

데이터의 shape, pixel, 차원수, 이미지 확인

이미지 데이터를 일일이 옮기기 힘드니 numpy 배열 파일로(.npy) 저장

* 모델링 과정

Conv2D, MaxPooling2D, Dropout, Flatten, Dense 레이어 사용하여 CNN 모델 설계

훈련 데이터를 CNN 모델에 넣고 학습시킴

Early Stopping 방법을 사용하여 6번의 진행동안 loss값이 갱신되지 않으면 훈련을 멈춘다.

Accuracy, Loss 그래프 확인

성능이 가장 좋은 순간을 ModelCheckpoint를 통해 그 순간의 모델을 파일로(.h5) 저장

새로운 신발 사진을 임의로 인터넷에서 크롤링해온 후 제대로 예측이 되는 지 확인

3) 텍스트 데이터 - 수집, 정제

* 데이터 수집 출처

무신사스토어 홈페이지 -> 스니커즈 & 신발 카테고리 -> 우리가 지정한 카테고리에 맞는 카테고리 설정 -> 상품평순 -> 상위 20개의 상품에서 일반 후기 최대 10페이지 크롤링

* 데이터 수집 과정에서 발생한 문제점

상품평 순으로 한 상위 20개의 신발의 리뷰를 하는 건데도 리뷰가 없는 신발이 있었다.

한 사람이 동일한 리뷰를 10개 이상 올린 경우가 있었다.

* 해결 방안

리뷰가 없는 신발의 공식 브랜드 홈페이지에 가서 후기를 크롤링해왔다.

텍스트 전처리 과정에서 중복 데이터를 제거하도록 코딩하였다.

* 분석 과정
  + 중복 데이터, 결측값 제거
  + 특수기호, 이모티콘, 숫자 제거 (한글, 띄어쓰기만 남기기)
  + 띄어쓰기, 맞춤법 교정
  + 토큰화
    - Mecab
    - Okt
    - 다른 라이브러리를 사용하지 않은 이유 : 속도가 너무 느리고 성능도 Mecab, Okt와 비슷하기 때문이다.
    - 일반적인 토큰화 방법을 사용하지 않고, 각 라이브러리의 품사 홈페이지에 들어가서 명사, 형용사, 부사만 토큰화 하도록 품사의 형태를 지정해주었다.
  + 불용어 제거
* 토큰의 최대 길이 정하기
* 단어 사전 만들기

단어의 빈도수 높은 순으로 나열

제일 많이 사용된 단어 10개를 해당 카테고리(‘브랜드-종류’)의 추천 태그로 사용하기로 했다.

* 최종 데이터프레임에 카테고리와 추천 태그를 미리 넣어놓는다.

4) 결과 확인 및 모델 수정

* 새로운 테스트 이미지를 함수에 집어넣게되면, CNN 모델을 통해 해당 신발의 카테고리(‘브랜드-종류’)를 예측하고, 그 예측결과에 맞는 추천태그를 데이터프레임에서 찾아서 결과적으로 신발에 해당되는 카테고리와 추천태그를 출력해준다.

* 이미지 CNN 모델링
  + 메인 사진 외에 세부 사진비율이 더 많아서 예측률과 로스율이 떨어지는 결과가 발생했다.
  + 추가적으로 전이 학습 방법 이용 -> 다른 사람들이 미리 학습해놓은 모델 이용 -> 모델의 정확도가 높아지기를 기대 -> 오히려 이전보다 성능이 안 좋게 나왔다. -> 전이 학습 방법을 사용하지 않겠다.
* 텍스트 전처리
  + 두 가지 라이브러리를 사용했는 데 Okt가 우리가 원하는 결과에 더 가까운 글자의 형태를 나타내줘서 Okt를 사용하기로 했다.
  + 일반적인 불용어 뿐만 아니라, 신발을 분석하는 것이니까 신발의 브랜드 이름, 신발의 종류, 기타 불용어들을 더 추가하였다.

5) 최종 결과 도출