

중고거래 플랫폼 내 가격 예측, 추천 시스템, 판매 확률 예측

KREAM과 당근마켓 플랫폼을 중심으로

2020131302 하진우
2021320322 윤민서

Contents •

프레젠테이션 목차

Contents 01

가격 예측

Dataset

EDA

Modeling

Result

Contents 02

추천 시스템

Dataset & Structure

EDA

Modeling

Result

Contents 03

판매 확률 예측

Dataset

EDA

Modeling

Result

Contents •

2차 발표 요건

주제 발표, 분석 목표

데이터 분석, 전처리

가설



2차 발표 내용

Dataset

EDA, Modeling

Result, Develop

가격 예측

1. Dataset

가설

- 중고 플랫폼의 게시글을 통해 가격 예측
- 게시글 내용 중 아래 주요하게 고려
 - 브랜드
 - 상품명
 - 카테고리
 - 관심도
- 선행 연구에서 활용한 RANDOM FOREST 와 LINEAR REGRESSION 테스트

당근마켓

- 게시글 10만개 크롤링
- 변수
[TEMPERATURE, TITLE, CATEGORY, TIME POSTED, PRICE, DESCRIPTION, INTEREST, CHAT, VIEW]
- RANDOM FOREST 모델링
- MSE: 14278428522.93
- R SQUARED: 0.13

해당 데이터로 가격 예측 어렵다고 판단

가격 예측

1. Dataset

KREAM

- KREAM 공식 페이지
- 카테고리별 TOP100 상품 700개 크롤링
- 변수
 - LINK
 - TRADE
 - BRAND
 - NAME(KOR / ENG)
 - INTEREST
 - REVIEW
 - PRICE 1~5

KREAM

BT21 추천 랭킹 럭셔리 남성 여성 발견

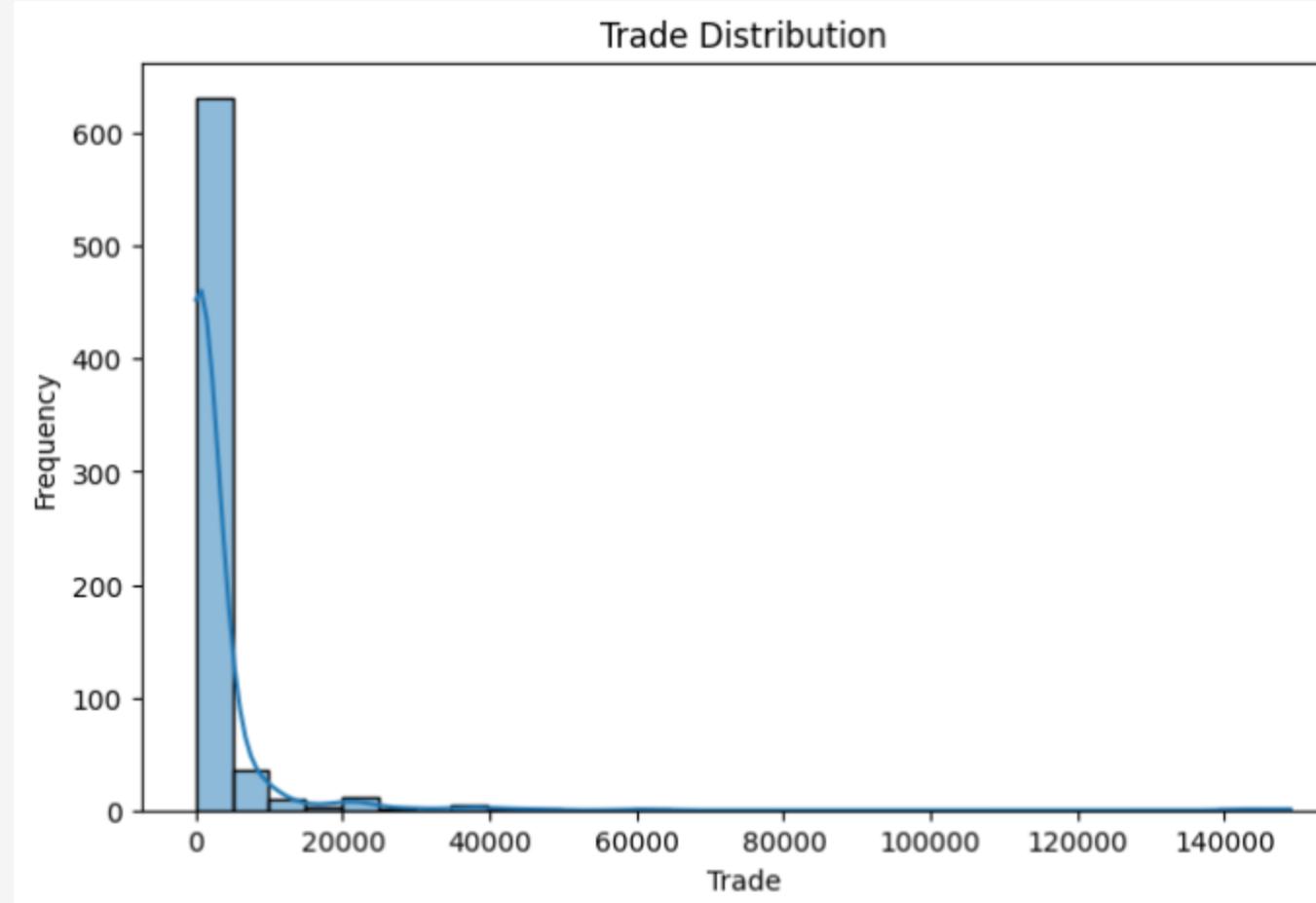
남성 신발 인기 순위

순위	상품 이미지	상품명	가격	배송
1		Nike Nike Air Force 1 Mid '07 White Wolf Grey	74,000원	빠른배송
2		Nike Nike Manoa Leather Haystack	83,000원	빠른배송
3		New Balance New Balance Puflee v2 Black	126,000원	빠른배송

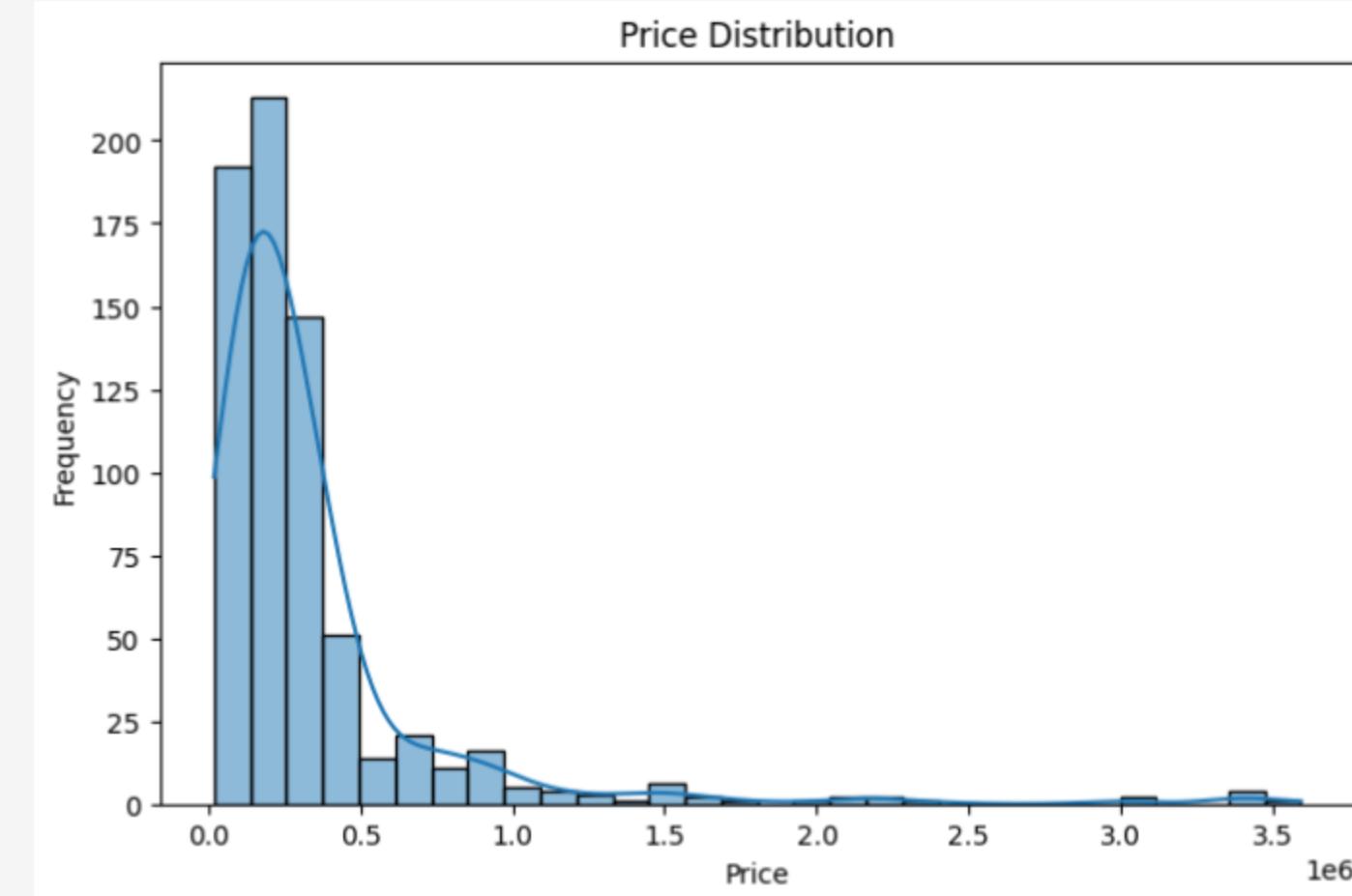
가격 예측

2. EDA

Trade



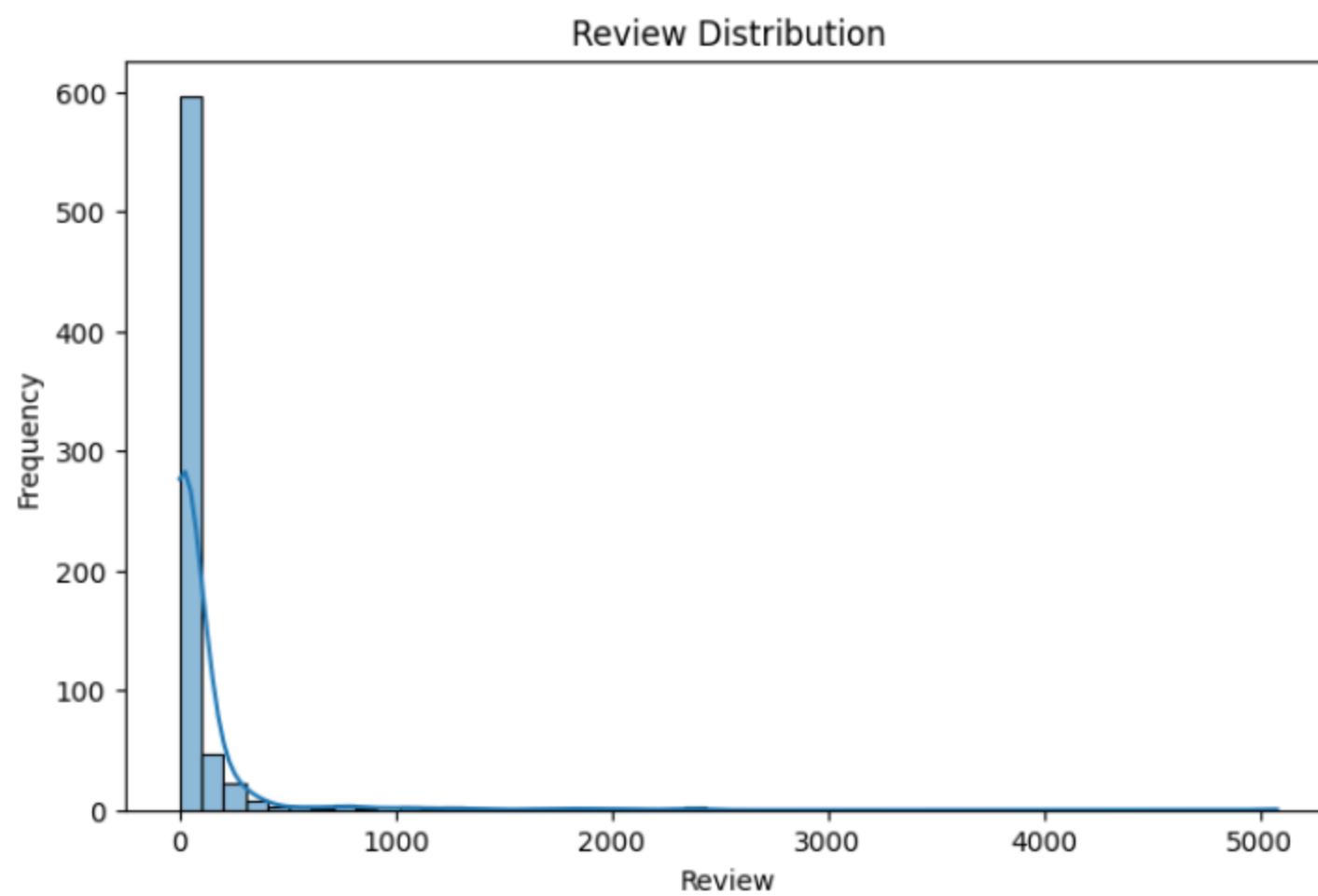
Price



가격 예측

2. EDA

Review



가격 예측

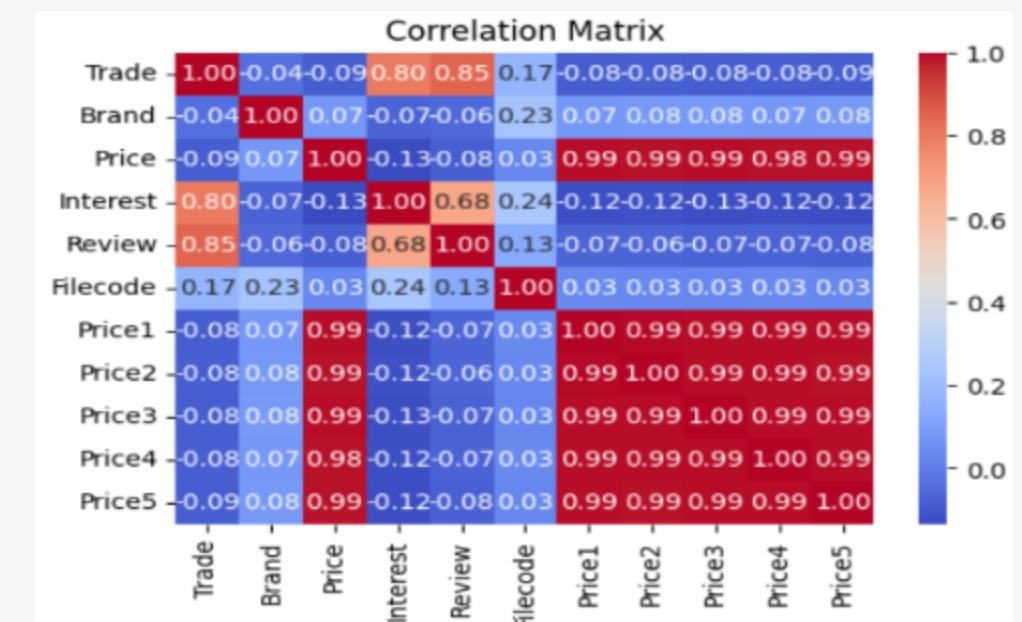
3. Modeling

1) 전처리

- Name(Korean), Link column drop
- Trade, Interest, Review column 내 문자 제거
- Brand column 라벨 인코딩
- 문자 변수 data type 정수/실수로 변환
- Price ~ Price 5 column의 결측치 최근 거래 가격(Price1)으로 대체

2) CORRELATION MATRIX

- 현재 가격(Price)과 높은 상관관계를 가진 변수 분석
- 현재 가격(Price)와 최근 5개 가격(Price 1~Price 5) 간의 높은 상관성 확인



가격 예측

3. Modeling

Random Forest

MEAN SQUARED ERROR: 4612943907.14

ROOT MEAN SQUARED ERROR: 67918.66

MEAN ABSOLUTE ERROR: 21128.0

R-SQUARED: 0.97

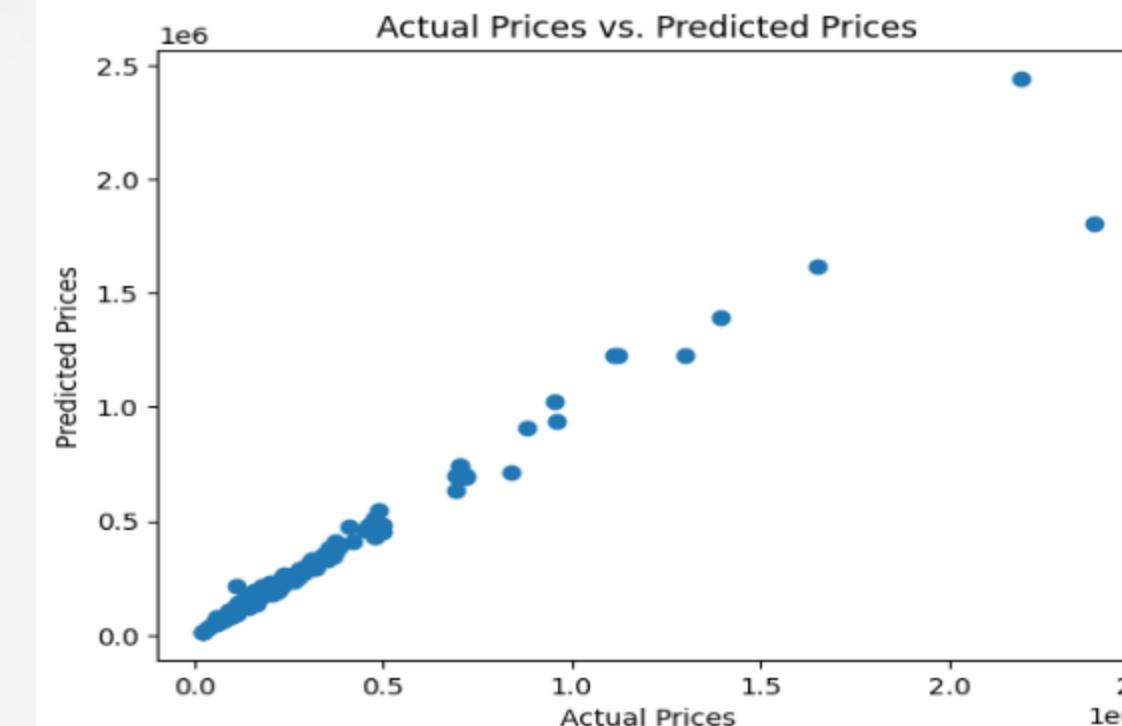
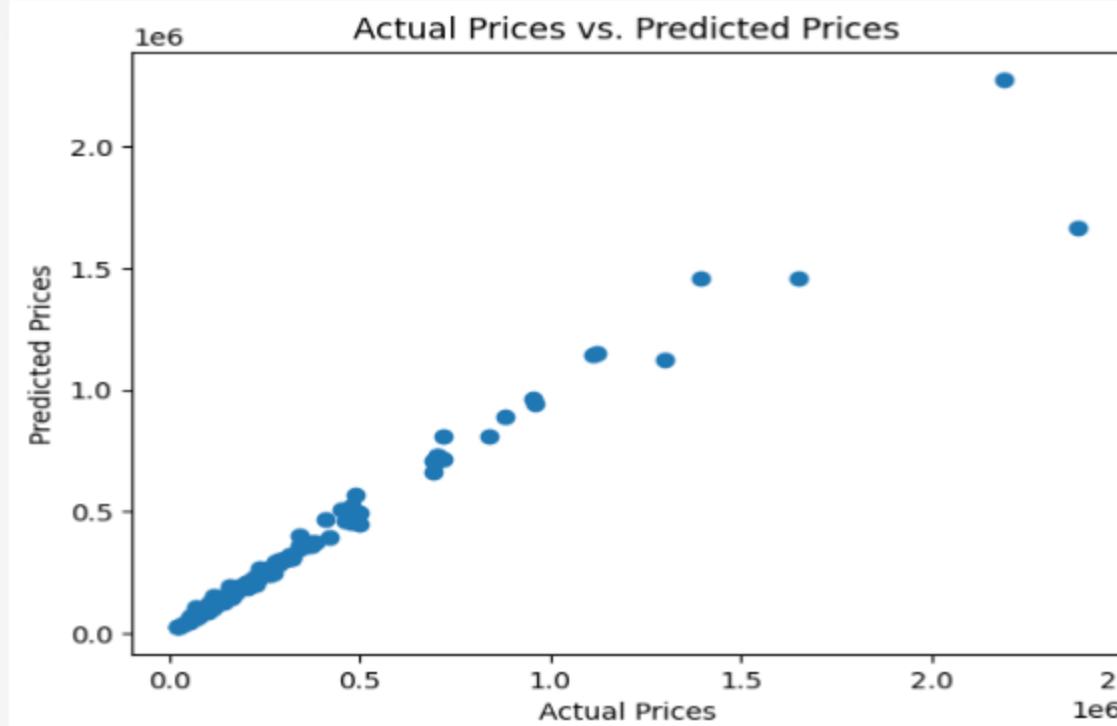
LINEAR REGRESSION

MEAN SQUARED ERROR: 3624270249.57

ROOT MEAN SQUARED ERROR: 60201.91

MEAN ABSOLUTE ERROR: 23757.12831365042

R-SQUARED: 0.97



가격 예측

4. Result



Result

- RANDOM FOREST, LINEAR REGRESSION 모두 우수한 성능 확인
- 과거 가격 데이터 바탕으로 현재 가격 예측 가능
- 과거 가격 데이터 이외 변수는 상관성 낮았음

Develop

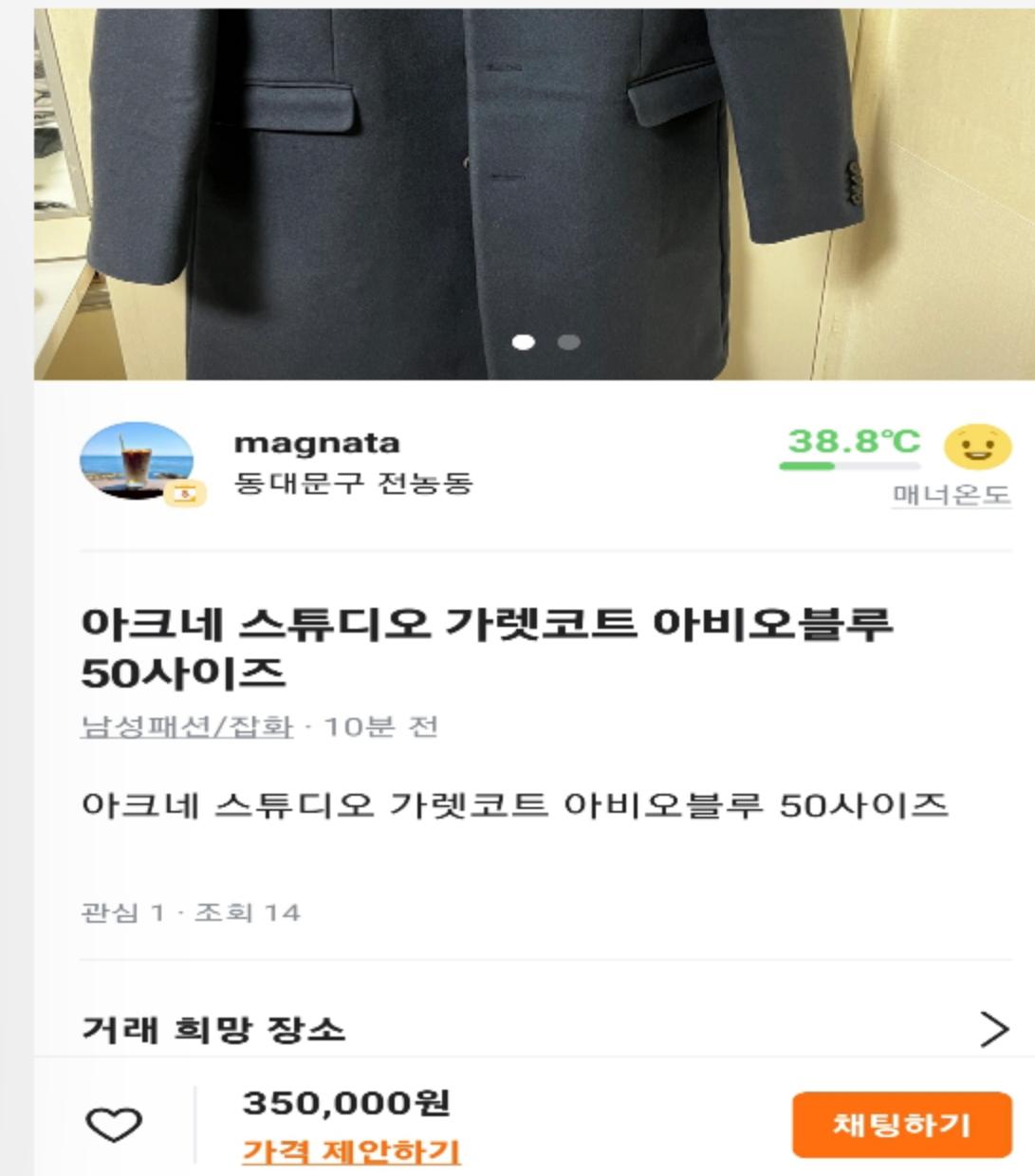
- 아이템 수 확대(700개 → 10000개)
- 과거 가격 데이터 확대(5개 → 30개)
- 외부 변수 추가(EX. 기간 내 포털 검색량)

추천 시스템 - 카테고리 간선 예측

1. Dataset & Data Structure

당근마켓

- 게시글 10만개 크롤링
- 변수
[TEMPERATURE, PRICE, INTEREST, CHAT, VIEW, UP(끌올), TIME]
- + WORD EMBEDDING
- LOGISTIC REGRESSION
- GCN, GAT, SEAL
- 새로운 모델 제시: LMKE + SEAL



추천 시스템 - 카테고리 간선 예측

1. Dataset & Data Structure

왜 일반적인 추천 시스템이 아닌가?

- 각 유저의 데이터 여러 개에 대하여 연결을
지은 그래프로 학습을 진행해야 함
- 그러나 유저 ID의 고유 번호에 대한 주소 접
근을 통해 크롤링을 하는 것이 불가능
- 유저 기반 추천 시스템이 아니라 카테고리
기반 추천 시스템으로 목적성 변경
- TASK를 추천 시스템과 비슷한 면이 있는
간선 예측으로 변경

작업 환경에 따른 모델 변동

- LIGHTGCN을 사용한다고 소개했음
- TASK가 변경되어서 GCN을 그냥 베이스
라인 중 하나로 선택
- 판매 확률 예측 TASK: 모델 별 대결 구도
- 판매 확률 예측 TASK에서는 LLM 기반 확
률 추론을 진행한다고 하였음
- 여기에서는 언어 모델의 규모를 줄이되 새
로운 데이터 구조와 추론 방식을 사용하도
록 결정 → LM + KG

추천 시스템 - 카테고리 간선 예측

1. Dataset & Data Structure

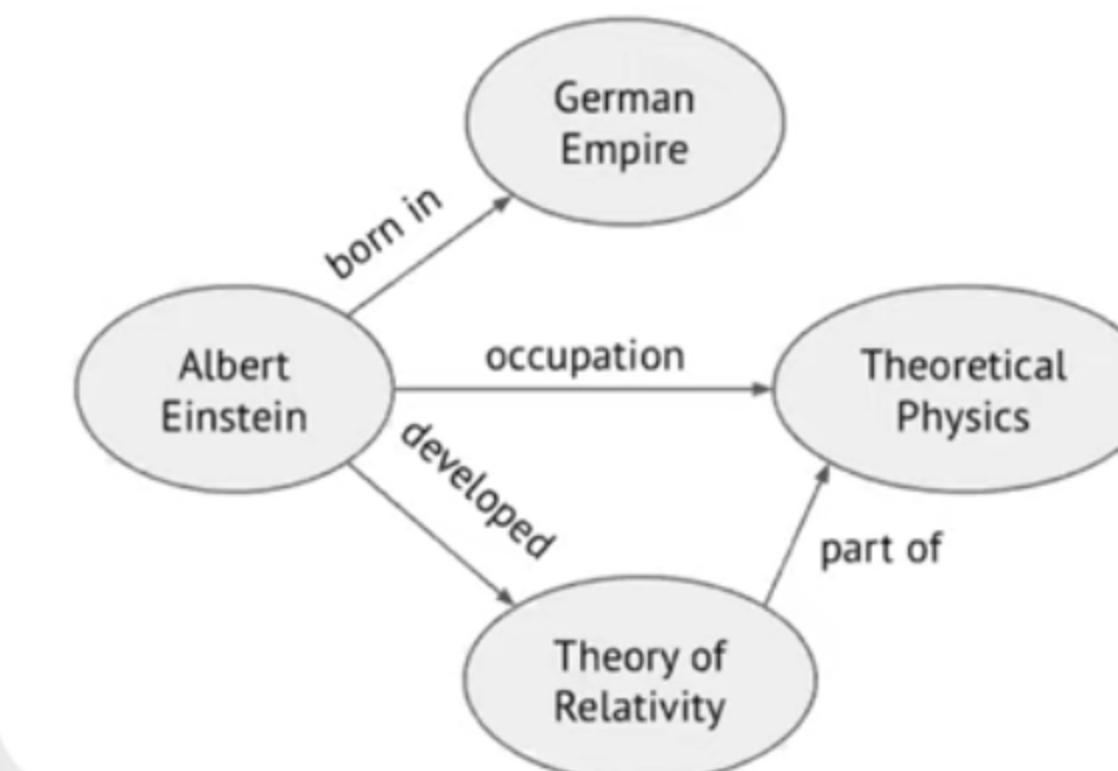
언어 모델 + 지식 그래프 구조

- LM + KG는 질문 답변, 이미지 설명 등의 TASK에도 널리 적용되고 있음
- 간선 예측에 사용되는 것은 그렇게 흔한 경우 아닙니다. 때문에 베이스라인 모델을 찾아서 모델을 재구축 한 후 추론을 진행

지식 그래프란?

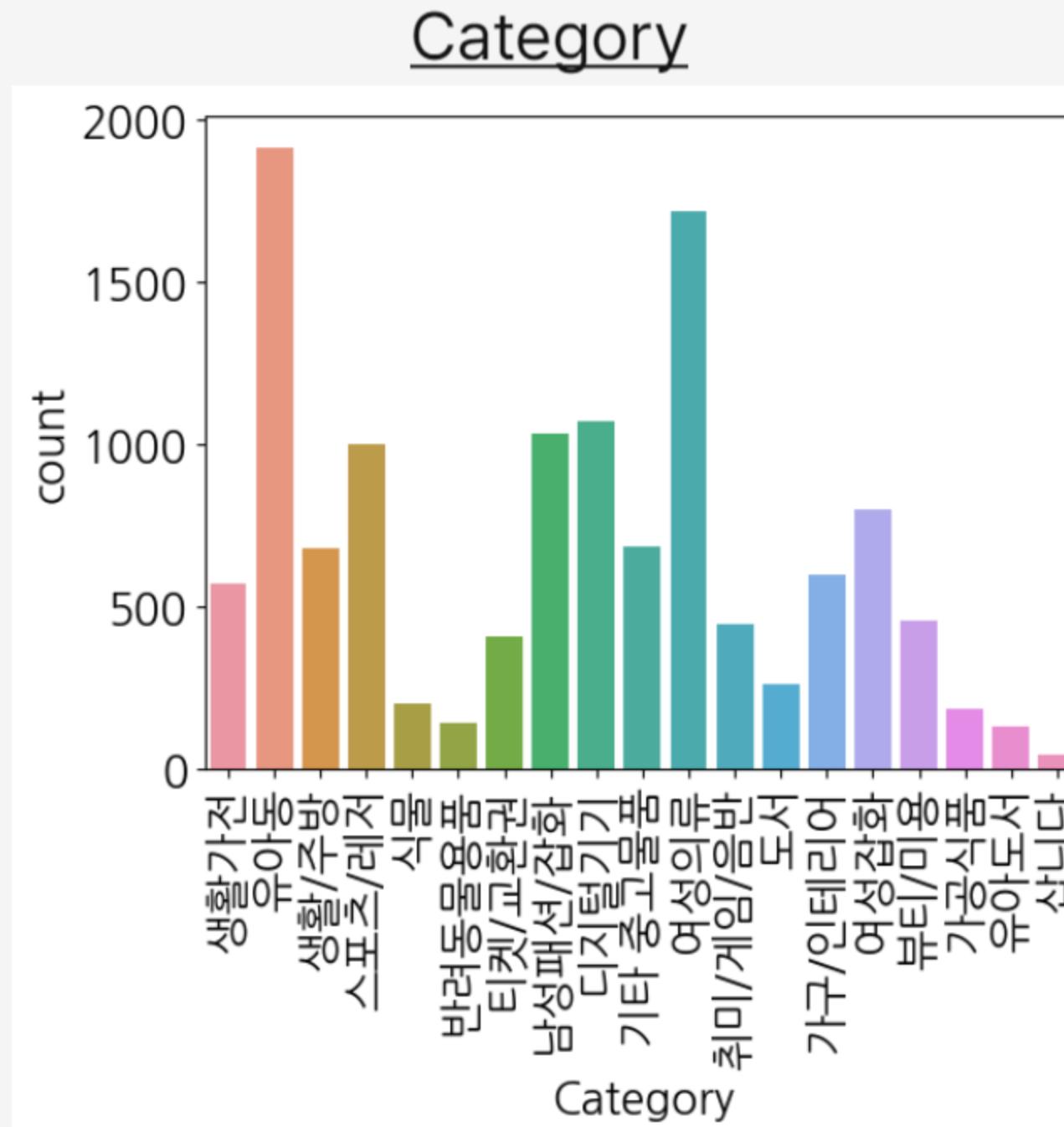
<HTTPS://BLOGS.SAP.COM/2020/08/13/WHAT-ARE-KNOWLEDGE-GRAPHS-AN-OVERVIEW/>

Albert Einstein was a German-born theoretical physicist who developed the theory of relativity.



추천 시스템 - 카테고리 간선 예측

2. EDA



- '삽니다' 카테고리는 DROP

추천 시스템 - 카테고리 간선 예측

2. EDA

간선 정의

- GPT-4에 프롬프트를 적절히 주면 표를 알아서 생성하게 할 수 있음
- GPT-4에게 배경, 목적 등을 설명
- 2개를 뽑는 모든 조합 중 연관성이 10점 만 점에 7점 이상인 것을 나열하라고 명령
- 연관성이 7점 이상인 두 카테고리에 대한 노드 사이에 간선을 연결
- 점수를 간선의 가중치로써 사용한다면 완전한 지식 그래프의 형태

카테고리1	카테고리2	관련성 점수
유아동	유아도서	9
유아동	여성의류	7
스포츠/레저	티켓/교환권	8
스포츠/레저	여성의류	7
남성패션/잡화	여성잡화	8
생활가전	디지털기기	8
생활가전	가구/인테리어	7
디지털기기	기타 중고물품	7
뷰티/미용	여성의류	8
뷰티/미용	여성잡화	7
가구/인테리어	여성의류	8
여성의류	여성잡화	9
취미/게임/음반	티켓/교환권	8
취미/게임/음반	디지털기기	7
생활/주방	가구/인테리어	8
식물	가구/인테리어	7
반려동물용품	기타 중고물품	8

추천 시스템 - 카테고리 간선 예측

2. EDA

데이터 불러오기

- 100,000개의 데이터로 그래프를 구축하면 COLAB의 RAM 용량을 초과하여 세션이 다운됨
- 100,000개 중에서 20,000개의 데이터를 사용 → 결측치 제거 후 12270개
- 한 에포크마다 약 2600만 개의 간선 중 6135개의 양성 샘플, 6135개의 음성 샘플을 추출하여 학습

Subgraph Visualization (1000 nodes)



추천 시스템 - 카테고리 간선 예측

3. Modeling

전처리

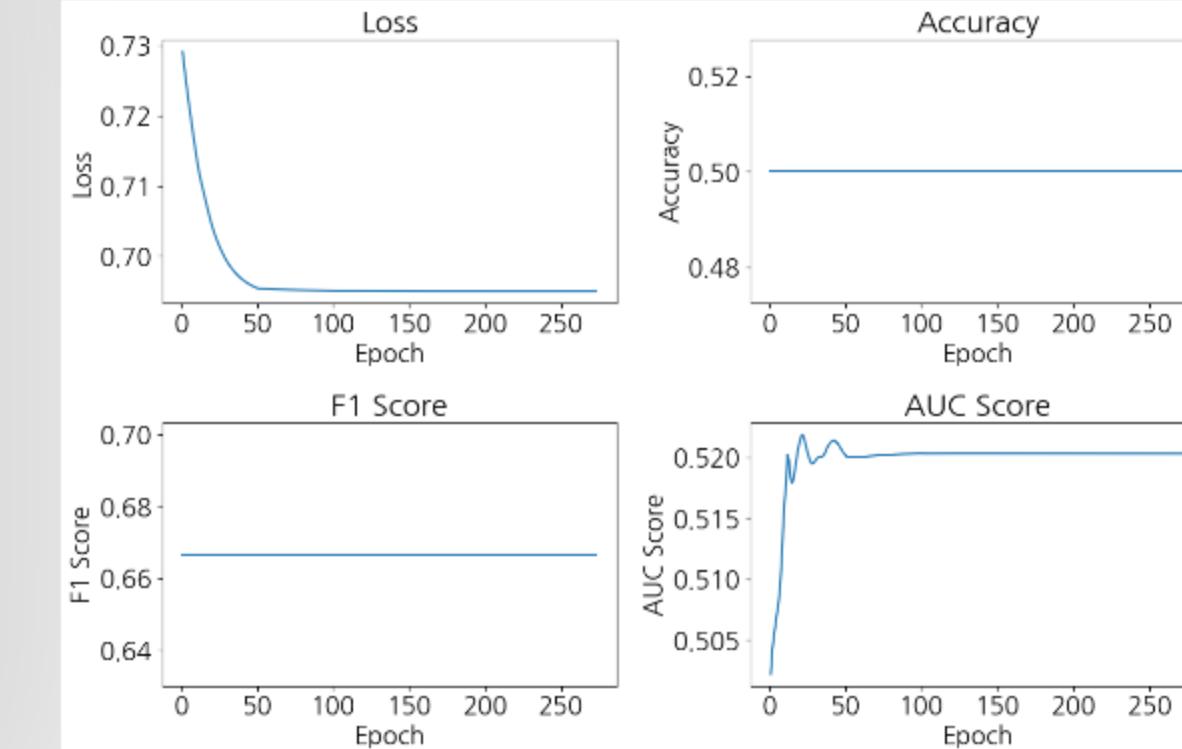
- 특성: [매너 온도, 가격, 관심, 채팅, 조회, 끌올, 업로드 이후 지난 시간, 제목]
- 최종 변수: [TEMPERATURE, PRICE, INTEREST_COUNT, CHAT_COUNT, VIEW_COUNT, UP, TIME, GLOVE_EMBEDDING * 100]
- 일반적인 특성 7개 + 제목을 100차원 단어 임베딩으로 정사영한 것
- 제목을 토큰화하여 각 토큰 100차원 임베딩으로 표현한 후 평균을 계산

LOGISTIC REGRESSION

ACCURACY: 0.5000

F1 SCORE: 0.6667

AUC: 0.5203



추천 시스템 - 카테고리 간선 예측

3. Modeling

GCN

ACCURACY: 0.5264

F1 SCORE: 0.5165

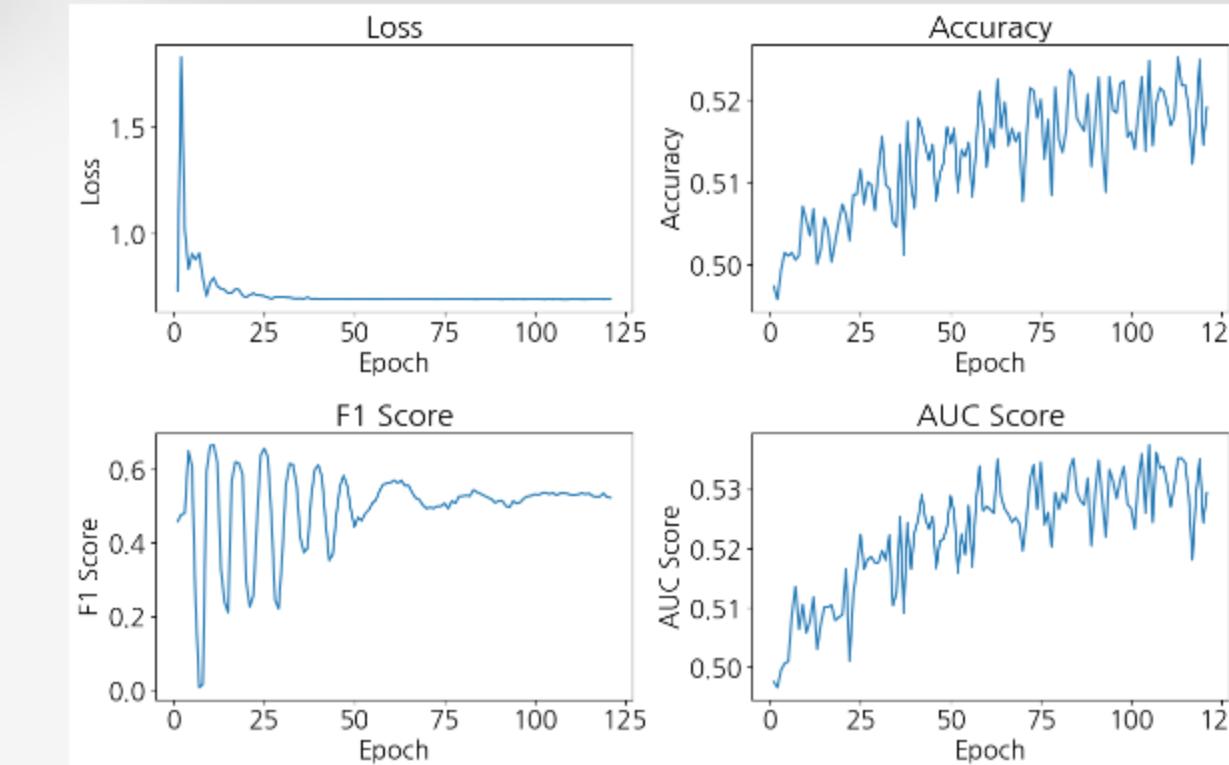
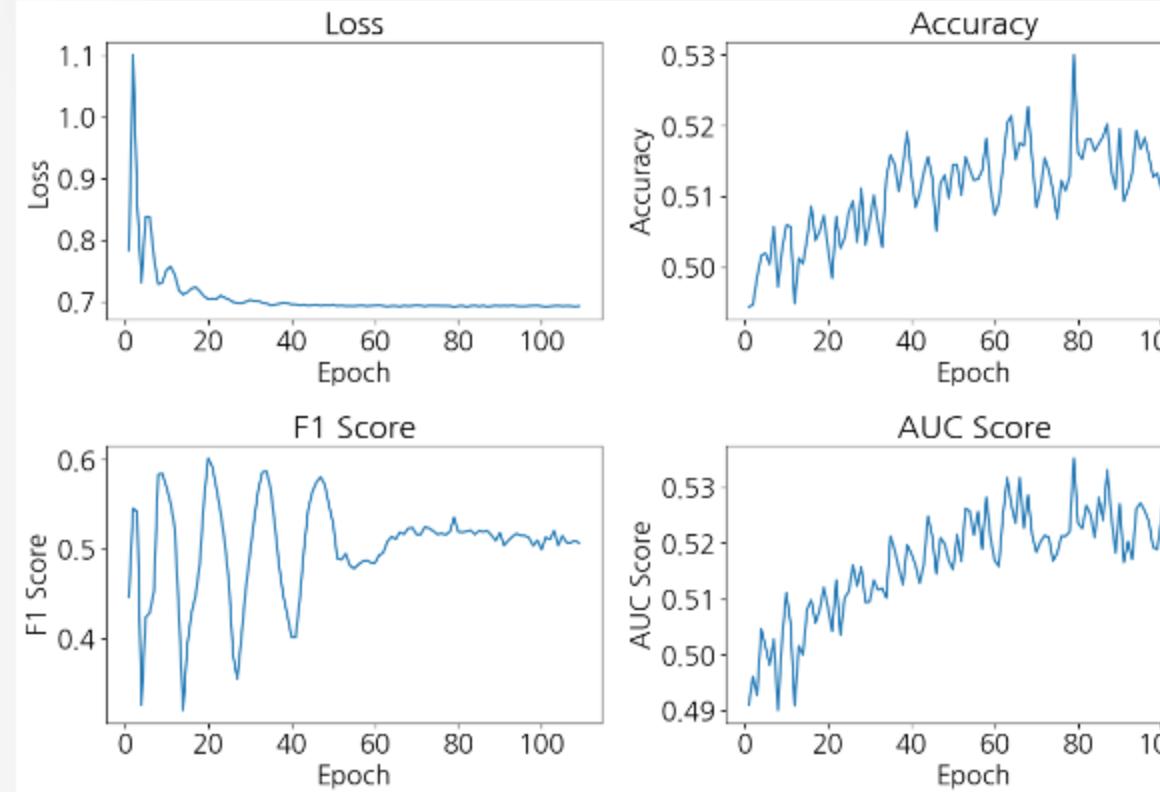
AUC: 0.5357

GAT

ACCURACY: 0.5251

F1 SCORE: 0.5393

AUC: 0.5442



추천 시스템 - 카테고리 간선 예측

3. Modeling

대표적인 간선 예측 모델 - SEAL

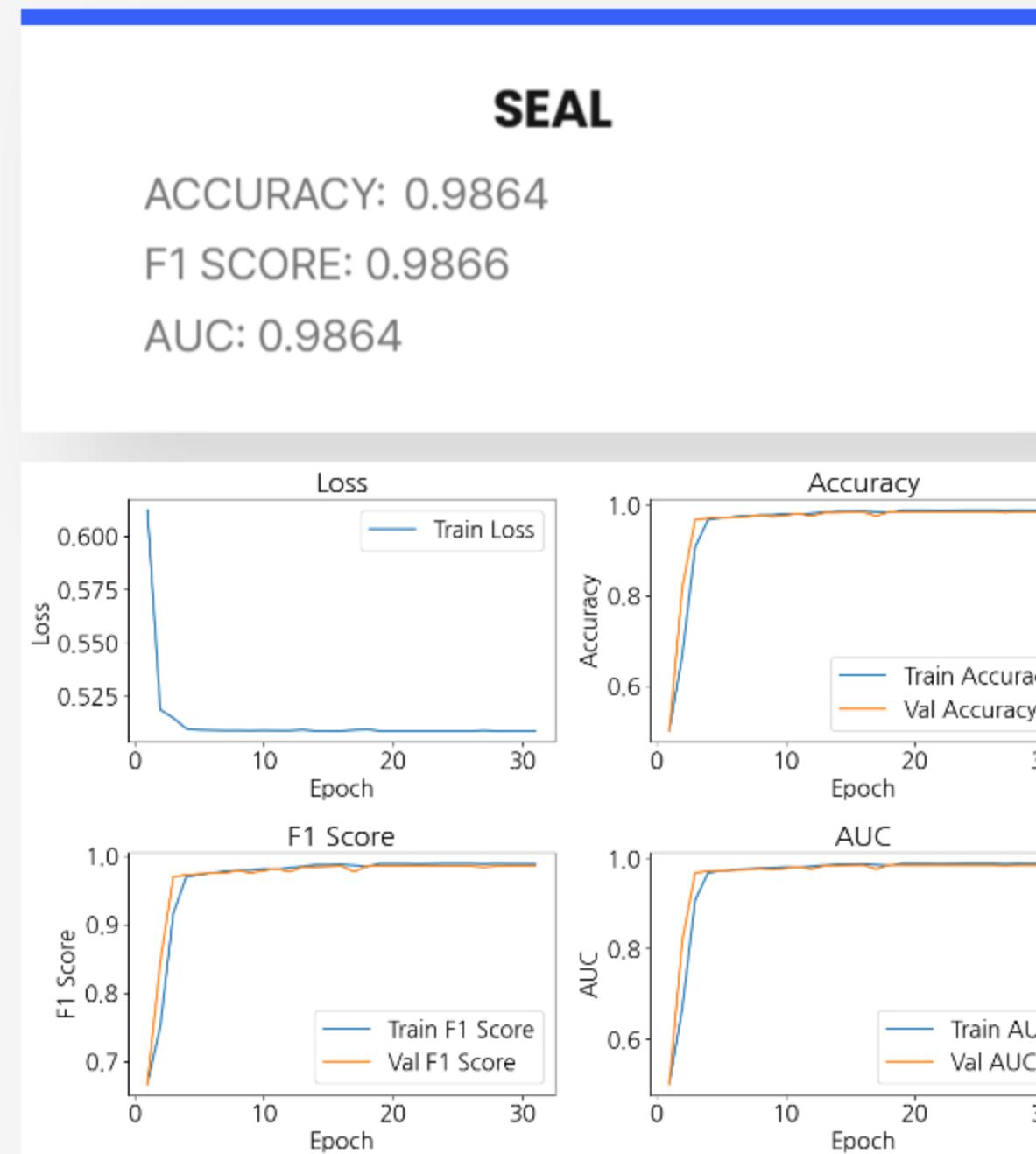
- 추천 시스템에서 GCN의 변형인 LIGHTGCN을 사용한다고 말했던 것을 유지하려고 노력
- 간선 예측 모형인 SEAL이 최종적으로 추론하는 방식도 GCN과 CONV1D로 구성

전처리 - SEAL

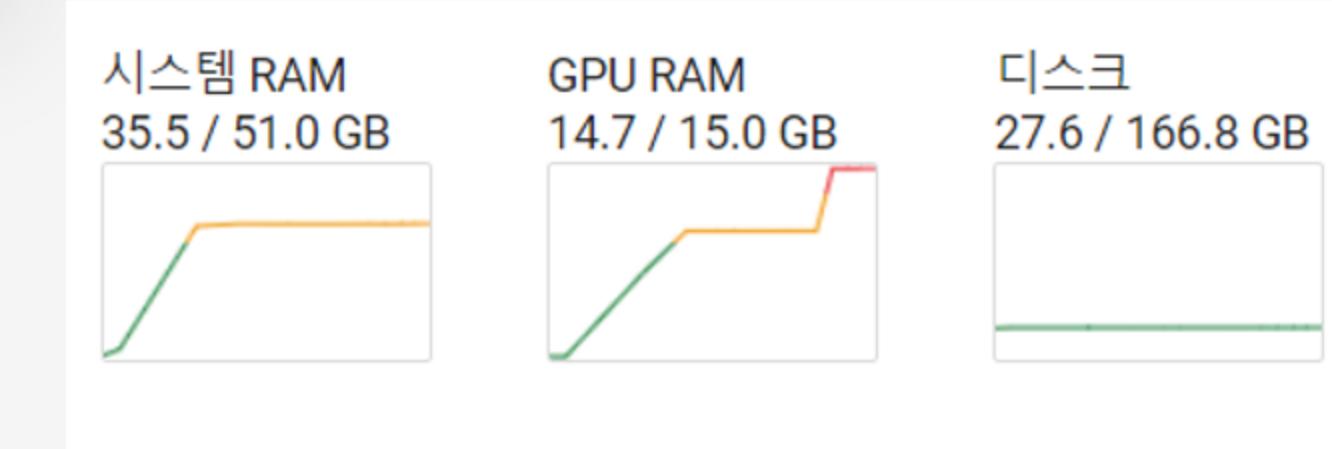
- 20,000개 전처리 → 약 30시간 (X)
- 3,000개를 서브샘플링 하여 전처리
- 그래도 COLAB 고용량 RAM(51GB)에서 실행이 되지 않아 간선 유무의 기준점을 7점에서 8점으로 상향
- TRAIN SET에 대하여 양성 샘플, 음성 샘플 각각 5000개를 추출
- TRAIN-VAL-TEST SPLIT 8:1:1
- VAL, TEST SET에 대하여 양성샘플, 음성 샘플 각각 625개씩 추출

추천 시스템 - 카테고리 간선 예측

3. Modeling



- 매우 뛰어난 성능이 관찰되었지만 학습 과정에 드는 GPU 메모리가 너무 크기 때문에 실제 서비스에 적용시키기에는 부적절하다고 생각



추천 시스템 - 카테고리 간선 예측

3. Modeling - Proposed Model: LMKE + SEAL

LMKE를 변형하는 아이디어

- LMKE라는 언어 모델 기반 지식 그래프 모델을 사용한 논문의 구조를 간선 예측 TASK에 맞게 변형하여 새로운 모델을 구축
- 모델 구조 면에서 LMKE에서 얻어온 것은 크게 없고 BERT의 EMBEDDING을 사용 한다는 흔한 아이디어를 가져옴

전처리 - LMKE + SEAL

- 앞서 사용했던 GLOVE의 100차원 임베딩 대신에 BERT의 121차원 임베딩을 사용
- 원래 변수 7개 + BERT 121개 = 128개: ATTENTION HEAD의 개수로 4, 8과 같은 숫자를 많이 쓰는데 이것으로 입력 차원을 나누었을 때 나누어떨어져야 함
- 그런데 BERT는 768차원: PCA를 사용하여 121차원으로 차원 축소

추천 시스템 - 카테고리 간선 예측

3. Modeling - Proposed Model: LMKE + SEAL

전처리 과정에서의 문제점

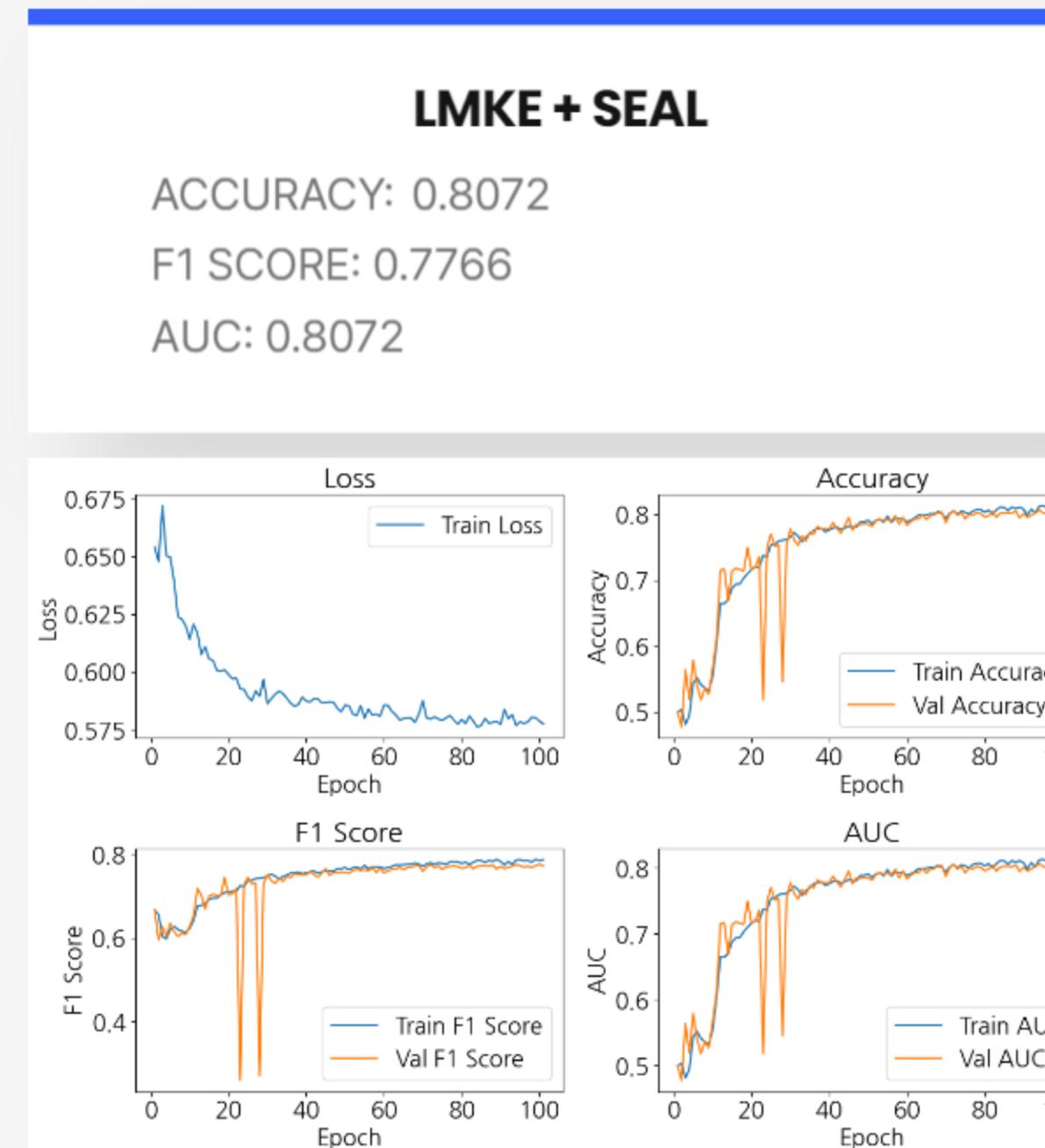
- 원래의 SEAL은 GCN LAYER 4개를 사용
- BERT 임베딩과 나머지 변수를 어텐션 메커니즘을 통해 학습하는 LAYER를 GCN LAYER 앞에 쌓았으나 성능이 좋지 않음
- GNN의 고질적 문제인 오버스무딩 때문이라고 생각함: GNN은 CNN처럼 LAYER를 많이 쌓기 쉽지 않음
- SEAL에 있던 GCN LAYER를 2개만 사용 했더니 성능이 크게 향상, 즉 오버스무딩이 맞았음

Oversmoothing?

- 노드들 간의 특성이 과도하게 혼합되어, 결국 서로 구분하기 어려운 유사한 특성을 가지게 되는 현상
- 특성들의 분포가 균등분포에 가까워짐
- GNN의 LAYER N개는 곧 N번 건너뛴 이후의 정보까지 고려한다는 것과 똑같은 의미 이기 때문에 LAYER가 지나치게 많아질 경우 지역적인 특성을 고려하지 못하게됨

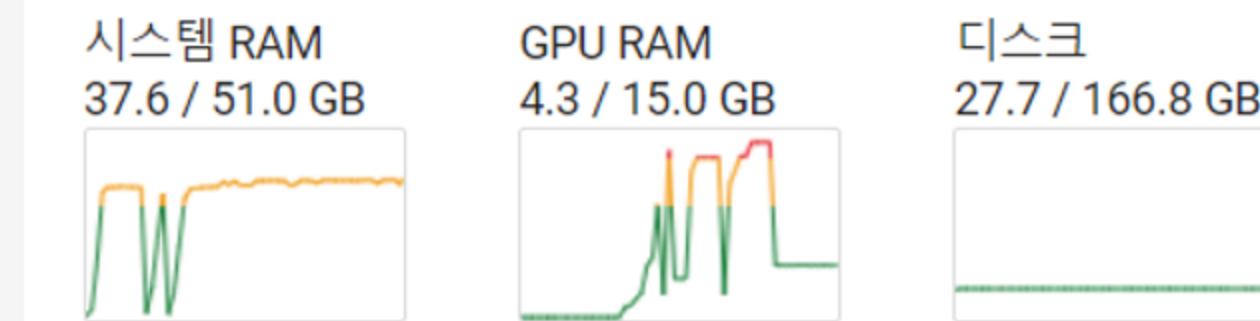
추천 시스템 - 카테고리 간선 예측

3. Modeling - Proposed Model: LMKE + SEAL



- LOSS가 조금씩 떨어지는 경향을 꾸준히 보이고 있으나 후반으로 갈수록 성능의 큰 향상은 없음
- 그러나 성능이 멈추지 않고 조금씩 올랐기 때문에 현재보다 높은 학습률에서 수렴 속도가 더 빠를 것을 기대
- 모델이 지역 최소를 찾으려고 시도할 때 가중치들이 잠시 한꺼번에 활성화되는 시점을 제외하면 70% 넘게 메모리를 절약할 수 있었음

Python 3 Google Compute Engine 백엔드 (GPU)
11월 5일부터 오전 1:08까지 리소스 표시



추천 시스템 - 카테고리 간선 예측

4. Result

아래 결과에 대한 시사점 / 가설

- 모델이 지역 최소를 찾으려고 시도할 때 가중치들이 잠시 한꺼번에 활성화되는 시점을 제외하면 70% 넘게 메모리를 절약할 수 있음
- 실제 서비스에서는 어떤 모델을 더 선호할 것인가?

- IT 기업을 운영하는 입장이라면 메모리가 많이 들지만 98%의 성능을 내는 모델을 사용할 것인가, 아니면 메모리가 30% 수준으로 들면서 80%의 성능을 내는 모델을 사용할 것인가?
- 또한 유저의 입장이라면 렉이 많이 걸리지만 98%의 정확도를 보여 주는 서비스를 사용할 것인가, 아니면 사용 환경이 쾌적하지만 80%의 정확도를 보여 주는 서비스를 사용할 것인가?

→ 상황과 목적에 따라 달라질 것

추천 시스템 - 카테고리 간선 예측

4. Result

실패 유형

- 사용자에게 추천되는 카테고리 중 잘못된 것은 위양성으로 분류한 케이스
- 본래는 설명력을 높일 수단으로 출력해 본 것이었으나 데이터 전처리 과정에서의 실수를 발견함
- 주체과 객체가 같은 경우를 양성으로 고려해 주지 않았고 이러한 경우가 1250개의 VALIDATION SET 중 32개가 있었음
- 완전히 불합리적으로 틀린 케이스는 거의 없음

False Positive Edge: 여성잡화 여성잡화
False Positive Edge: 남성패션/잡화 뷰티/미용
False Positive Edge: 여성잡화 여성잡화
False Positive Edge: 여성의류 여성의류
False Positive Edge: 여성의류 여성의류
False Positive Edge: 생활/주방 남성패션/잡화
False Positive Edge: 여성잡화 식물
False Positive Edge: 뷰티/미용 남성패션/잡화
False Positive Edge: 여성잡화 뷰티/미용
False Positive Edge: 남성패션/잡화 여성의류
False Positive Edge: 여성잡화 남성패션/잡화
False Positive Edge: 여성의류 여성의류
False Positive Edge: 여성의류 여성의류
False Positive Edge: 뷰티/미용 남성패션/잡화
False Positive Edge: 남성패션/잡화 뷰티/미용
False Positive Edge: 남성패션/잡화 뷰티/미용
False Positive Edge: 여성잡화 가구/인테리어
False Positive Edge: 여성의류 여성의류
False Positive Edge: 도서 여성잡화
False Positive Edge: 가구/인테리어 남성패션/잡화
False Positive Edge: 가구/인테리어 여성잡화

추천 시스템 - 카테고리 간선 예측

4. Result



Result

- SEAL이 가장 우수한 성능
- 제안하는 모델인 언어 모델 기반으로 지식 그래프의 간선을 예측하는 모델도 합리적인 성능을 보이며 컴퓨팅 자원 효율이 우수함

Develop

- 정확한 전처리가 된 데이터로 다시 실험한 결과는 3차 발표 때 반영
- 현재 트렌드인 LLM의 임베딩을 가져와서 간선 예측을 한다면 어떨까?
→ 나중에 GNN에 관심이 있는 기업이 해줄 것

판매 확률 예측

1. Dataset

당근마켓

- 게시글 10만개 크롤링
- 변수: [TITLE, DESCRIPTION]
- LINEAR, LOGISTIC REGRESSION
- RANDOM FOREST
- SVM
- TF-IDF VECTORIZER
- KULLM
- GPT-4

magnata
동대문구 전농동

38.8°C 😊
매너온도

아크네 스튜디오 가렛코트 아비오블루
50사이즈

남성패션/잡화 · 10분 전

아크네 스튜디오 가렛코트 아비오블루 50사이즈

관심 1 · 조회 14

거래 희망 장소 >

350,000원
가격 제안하기

채팅하기

판매 확률 예측

1. Dataset

Task 변동 사항

- 판매 완료 글에 접근이 되지 않음
(URL을 통해 들어가면 게시물 숨김 처리)
- 판매가 되지 않은 데이터들에 접근하여 크롤링을 진행
- TASK를 7일 이내 판매 여부 확률 예측으로 조금 변경
- 2차 발표: 7일 이내 판매 여부 예측
- 3차 발표: 7일 이내 판매 확률 예측
(판매 여부 예측의 성능이 잘 나온다면)

Train-Val-Test Split

- 50000, 10000, 309개
- LLM과 비교 분석을 하는 TASK를 진행한다고 했었는데, 학습에 비해서 평가가 상당히 오래 걸리는 LLM 미세 조정 기법의 특성 때문에 TEST SET을 작게 설정함
(TEXT GENERATION을 하나의 데이터마다 전부 해야 하니까)

판매 확률 예측

1. Dataset

본격적인 실험 전에

- 데이터셋 자체가 유효한지 알아보기 위해 거꾸로 제목, 상품 설명을 DROP하고 실험을 진행
- 데이터셋이 유효하다는 것이 무엇인가 → 데이터가 일관성이 없고 질이 좋지 않거나 학습에 충분하지 않은 양인지 먼저 확인
- 텍스트와 같이 복잡한 변수가 다 빠져 있는 아주 쉬운 TASK

예상 결과

- 업로드 이후 지난 시간과 끌올이 음의 상관 관계 다소 가지고 있음
- 업로드 이후 지난 시간과 관심이 음의 상관 관계를 약하게 가지고 있음 (EDA 파트)
- 50000개의 학습 데이터는 로지스틱 회귀 모델을 학습하기에 아주 충분하다고 생각했기 때문에 학습이 잘 될 것으로 예상했음

판매 확률 예측

1. Dataset

실험 결과 (로지스틱 회귀)

- TEST ACCURACY: 1.0000
- TEST F1 SCORE: 1.0000
- TEST AUC SCORE: 1.0000
- 그런데 성능이 생각보다 매우 잘 나와서 TASK를 살짝 어렵게 조정
- 7일 이내에 팔렸는지 여부를 맞히는 것이 아닌 며칠 동안 안 팔렸는지 맞히는 것으로 테스트를 진행함
- 모델: 로지스틱 회귀 → 선형 회귀

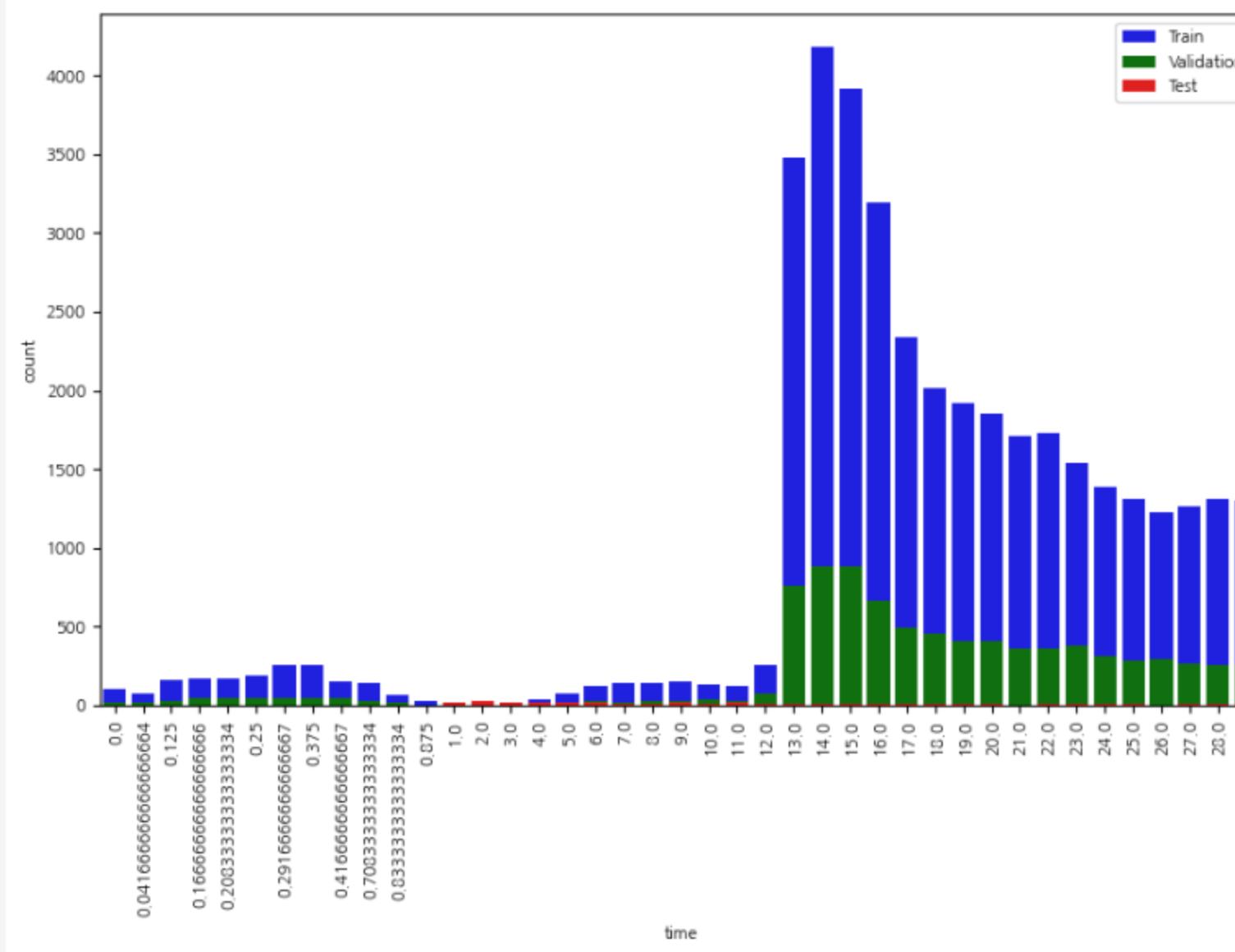
실험 결과 (선형 회귀)

- TEST MEAN SQUARED ERROR: 6.450727846383241E-29
- TEST R-SQUARED: 1.0
- 선형 회귀 또한 성능이 매우 잘 나와서 학습에 적합한 데이터셋으로 간주하고 계속해서 실험을 진행함

판매 확률 예측

2. EDA

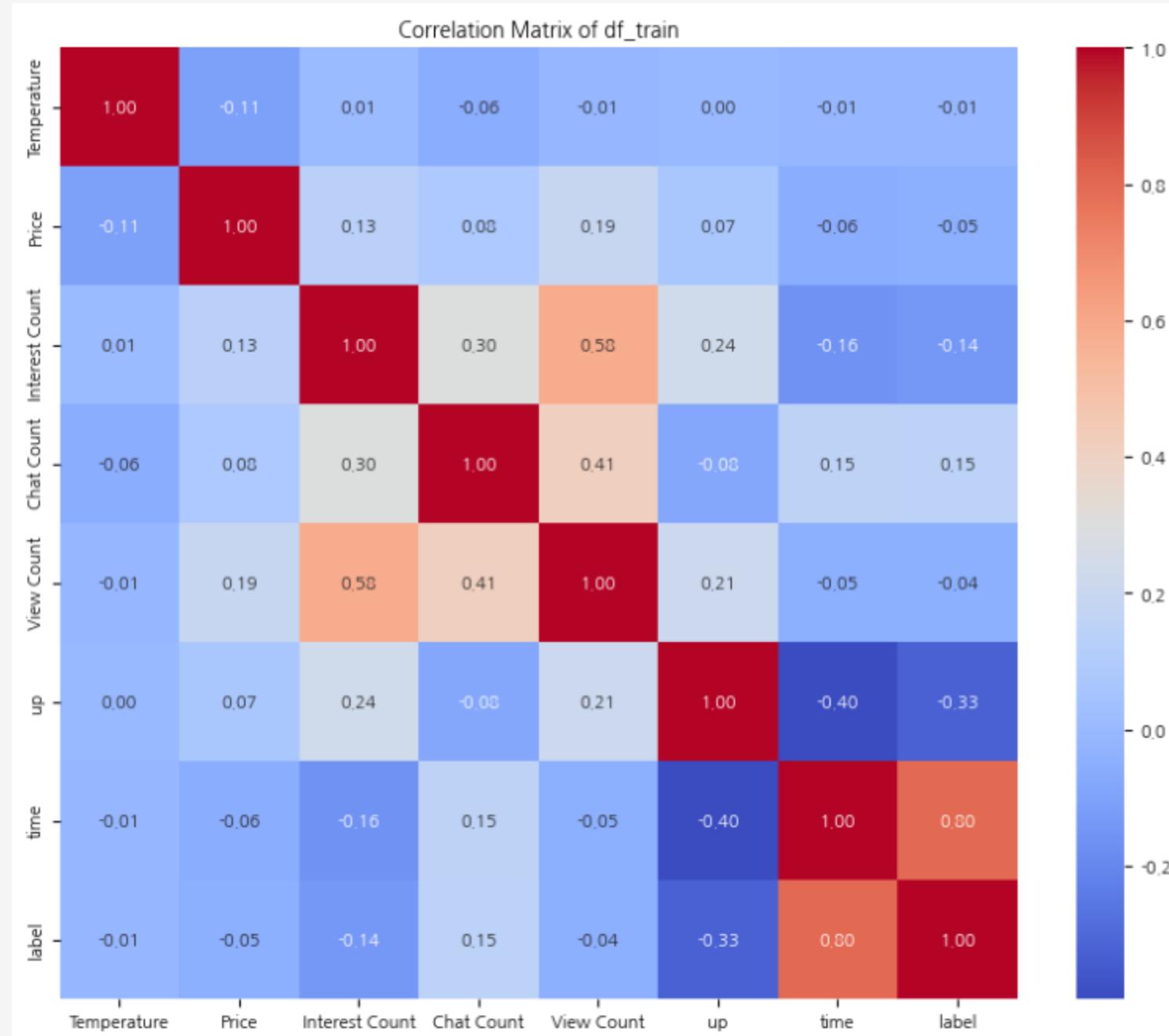
업로드 이후 지난 시간



판매 확률 예측

2. EDA

상관계수 행렬



- 본격적인 실험에서는 텍스트 데이터만을 사용할 것이기 때문에 큰 의미를 가지지 않음
- 데이터셋의 타당성을 확인하기 위한 예비 실험 과정에서 유의미한 결과를 제공

판매 확률 예측

3. Modeling

변수 선택

- 제목에 대한 텍스트를 통해 WORD2VEC 100차원 임베딩을 학습
- 상품 설명에 대한 텍스트를 통해 WORD2VEC 100차원 임베딩을 학습
- 최대 200개의 변수로 학습 진행 가능
- 그러나 중복되는 특성을 줄이고 학습 속도를 향상하고자 PCA를 통해 80%의 분산을 보존하도록 설정

변수 선택 결과

- 분산 비율 80%: 13개의 새로운 변수
- 95%의 분산도 보존하도록 해봤으나 유의미한 성능 차이가 나지 않아 학습의 효율성을 위해 80%로 결정
(95%의 경우 27개의 새로운 변수)

판매 확률 예측

3. Modeling

- 엄밀하게 보면 TRAINED와 N-SHOT이 양립하는 말은 아님
→ PROMPT의 SHOT 숫자는 PROMPT에 직접적으로 넣어 준 예시의 개수를 뜻하는 것으로 간주

Method	Accuracy	F1 Score	AUC Score
Logistic Regression	54.69	68.75	54.84
Logistic Regression (PoV: 95%)	55.99	69.51	54.35
Random Forest Classifier	56.96	71.76	51.35
Random Forest Regressor	58.25	73.62	-
TF-IDF Vectorizer + Logistic Regression	64.52	77.55	-
TF-IDF Vectorizer + SVM	62.90	76.77	-
TF-IDF Vectorizer + Random Forest Classifier	54.84	66.67	-
KULLM-5.8B (baseline)	49.19	52.57	-
KULLM-5.8B (trained 2-epoch, zero-shot)	53.40	68.28	-
KULLM-5.8B (trained 2-epoch, 2-shot prompt)	58.25	<u>73.62</u>	-
GPT-4 (prompt tuning)	<u>58.90</u>	72.57	-

판매 확률 예측

4. Result

결과에 대한 시사점 / 가설

- 기대했던 성능과는 거리가 멀다 (LLM)
→ 작은 모델을 사용해서일 수도 있고 아직 최적의 프롬프트를 찾지 못했음
- 프롬프트가 너무 길기 때문에 LLM이 후반부 맥락을 이해하지 못하는 경향이 있었지만 짧은 프롬프트를 사용하면 모델이 TASK 자체를 이해하지 못하는 양자택일 문제
- 앞으로 시도해 볼 방향성 중 파라미터의 크기와 프롬프트 중 어떤 것이 더 큰 영향을 미칠까?

너는 지금 중고거래 플랫폼을 운영하고 있는 데이터 분석가야.

아래에 입력되는 제목과 상품 설명을 참고해서 이 상품이 플랫폼에 업로드되고 나서 지난 시간이 7일 이상일지 또는 7일 미만일지 예측해 줘.

간단하게 생각하면 상품이 판매되는 데에 7일 미만이 걸릴지 또는 7일 이상이 걸릴지 예측하는 것과 비슷해.

네가 출력해야 하는 것은 상품이 플랫폼에 업로드되고 나서 지난 시간이 7일 이상일지 또는 7일 미만일지 예측한 결과야.

네가 지켜야 할 규칙이 있어. 규칙은 아래와 같아.

7일 이상으로 예측했으면 숫자 "1"을 출력하고, 7일 미만으로 예측했으면 숫자 "0"을 출력해야 함.

두 가지 예측 결과가 모두 존재하지 않아야 함. "1"과 "0" 두 개가 출력에 모두 포함되는 경우가 없어야 함.

예측 결과를 두 번 이상 출력하지 않아야 함. "1" 또는 "0" 딱 한 글자만 출력해야 함.

판매 확률 예측

4. Result

실패 유형

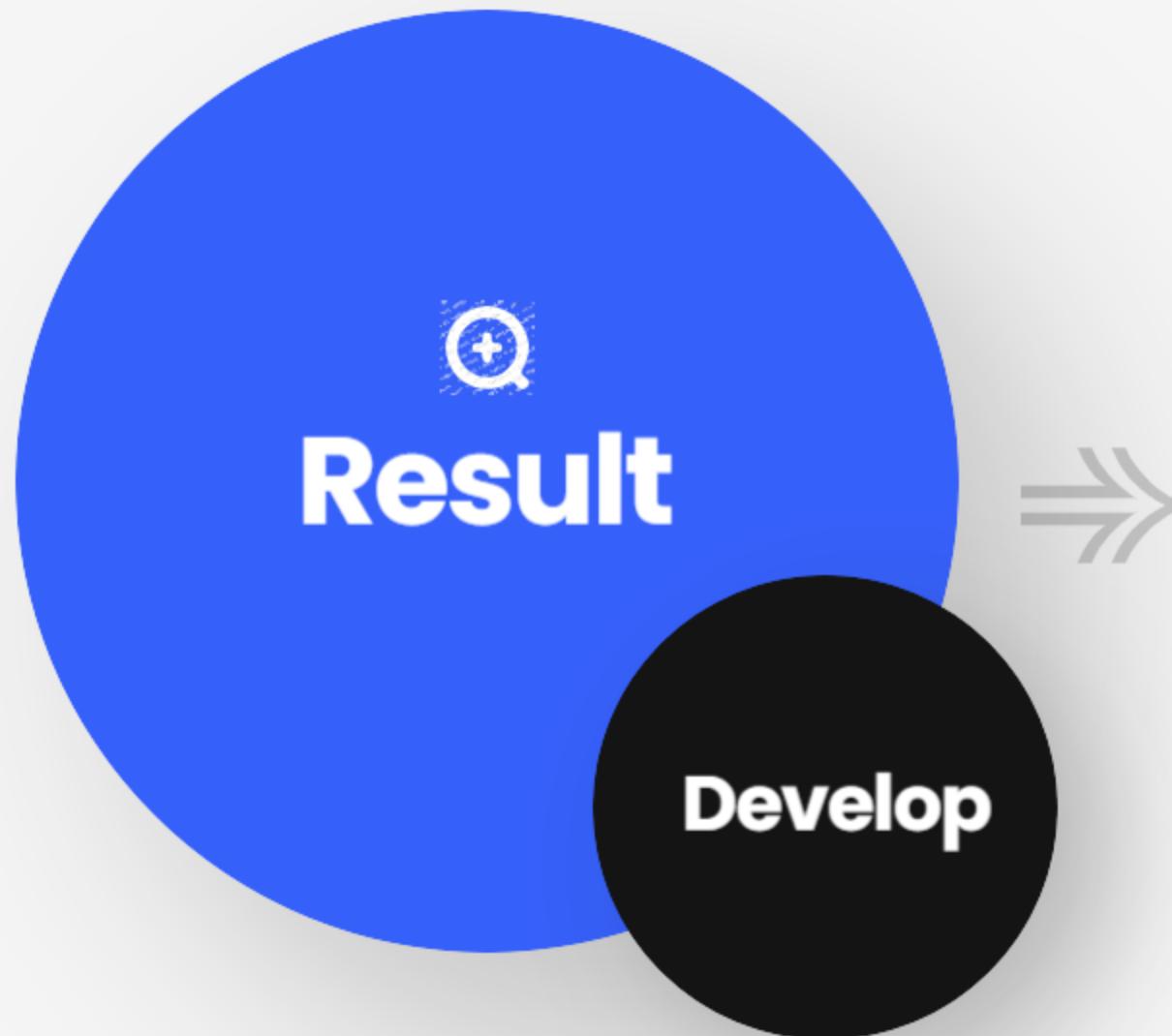
- 0 또는 1만을 출력해야 한다는 명령을 이해
하지 못하는 듯
- 라벨과 프롬프트를 바꿔서 다양한 시도를
해야 할 것으로 예상

예측 결과:

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17

판매 확률 예측

4. Result



Result

- 고전적인 자연어처리 방법 중 하나인 TF-IDF를 이용하는 것이 가장 우수한 성능
- LLM의 TASK 이해도가 그렇게 높지 않음

Develop

- LLM에 대해 더욱 다양한 시도를 할 필요가 있음
- LLM뿐만 아니라 LM + 추가 LAYER 구조도 시도해 볼 수 있음
(ADAPTER 계열)

Thank you

