AI 사서

소프트웨어학과 정찬호 201521001

소프트웨어학과 서상원 201720750

소프트웨어학과 성진호 201820808

소프트웨어학과 이동현 201920811

소프트웨어학과 김관주 202022307

**Introduction**

Background

도서관에서는 도서하고자 하는 책들을 DDC(듀이십진분류법)에 기반해 책의 장르를 나누고 그에 맞는 청구기호를 매겨 분류하고 있다. 그리고 분류된 청구기호에 따라 도서관의 일정한 위치에 책을 배치해 책을 쉽게 찾을 수 있도록 하고 있다.

다음은 청구기호를 부여하는데 쓰이는 DDC(듀이십진분류법)의 십진분류표의 일부이다. 청구기호에 책의 장르 정보가 포함되어 있음을 알 수 있다.

Assumptions, Requirements, Risks and Constraints

본 프로젝트는 책의 제목에 사용된 단어들은 책의 내용 및 장르를 함의하고 있음을 전제로 하고 있다. 우리는 기계학습을 통해 책의 제목에서부터 책의 장르를 유추해낼 수 있는가를 기존 도서 데이터를 통해 알아보고자 한다.

이를 위해서는 도서관에 있는 책들의 제목 및 청구기호 정보가 필요하다. 그런데 전제와 다르게 제목만으로는 책의 장르를 유추하기 쉽지 않을 수 있다. 이에 우리는 책의 목차를 추가로 수집하여 활용할 예정이다.

Project Goals and Success Criteria

기본적인 프로젝트의 목표는 새로 발간된 책이나 기존에 도서관 DB에 등록되어있지 않은 데이터를 넣었을 때 전문가에 준하는 정확도로 분류해낼 수 있는 것이다.

추가적으로 이번 프로젝트를 통해 책을 쓰는 사람은 책 제목에 어떤 단어를 써야 사람들에게 내용을 잘 전달할 수 있을지 알 수 있게 되고, 책을 분류하는 사람 및 읽을 책을 찾는 사람은 책의 제목만으로도 어떤 내용일지 예측할 수 있게 되고, 도서관 내에서의 책의 위치도 예측할 수 있게 된다.

성공의 기준으로는 테스트 정확도 80%를 넘는 것을 목표로 하고 있다.

**Data Preparation**

Data Explanation and How We Collected the Data

크롤링을 통해 아주대 중앙도서관 페이지에서 책들의 제목, 소장 위치, 청구 기호 등이 담긴 목록을 수집했다. 다음은 중앙도서관 홈페이지에서 예시로 다운받은 파일의 일부이다.

| **도서명** | **저자** | **출판사** | **출판년도** | **소장위치** | **청구기호** |
| --- | --- | --- | --- | --- | --- |
| Stochastic processes in engineering systems | Wong, Eugene | Springer-Verlag | 1985 | 중앙도서관 4층.자연 | 519.202462 W872ss |

데이터 중 도서명, 청구기호가 학습에 사용될 예정이다. 청구기호는 장르 정보가 포함된 앞 2자리만 사용하였다. 아주대학교 중앙도서관은 기존 10개 분류번호에 분류번호 65(경영학)을 추가한 변형된 버전의 듀이 십진법을 사용하고 있다. 즉, 도서들은 총 11가지의 장르로 분류되어 있으며, 우리는 이후 프로젝트 진행에서 아래 표와 같은 인덱스 번호를 분류에 사용했다.

| **분류번호** | **장르** | **분류번호** | **장르** | **분류번호** | **장르** |
| --- | --- | --- | --- | --- | --- |
| 0 | 총류 | 4 | 언어 | 8 | 예술 |
| 1 | 철학 | 5 | 자연과학 | 9 | 문학 |
| 2 | 종교 | 6 | 기술과학 | 10 | 역사 |
| 3 | 사회학 | 7 | 경영학 |  |  |

목차의 경우, 아주대 중앙도서관에서 사용하고 있는 알라딘 목차 api를 활용하여 크롤링을 진행했다. 다만 IP당 하루 쿼리 5000개의 횟수 제한이 있었고, 목차 정보가 제공되지 않는 책들이 많아 충분한 양의 목차를 확보하기가 어려웠다.

Data Cleaning and Pre-Processing

데이터 중 크롤링에 실패하여 오류가 난 데이터는 모두 제외하였다. 또한 양식에서 벗어난 데이터는 최대한 교정하였으나, 교정이 불가능 한 경우 제외하였다.

청구번호의 앞부분을 가지고 장르에 해당하는 Int 형식으로 변환시켜서 Label 역할을 할 수 있도록 처리하였다.

다음과 같은 정규표현식을 사용하여 한글이 최소 1글자라도 사용되며, 나머지 글자들 모두 한글, 영문, 숫자인 경우에는 koBERT, 아닌 경우 다국어 BERT 모델을 통해 학습시켰다.

[0-9|A-Z|a-z|ㄱ-ㅎ|ㅏ-ㅣ|가-힣|\s]\*[ㄱ-ㅎ|ㅏ-ㅣ|가-힣][0-9|A-Z|a-z|ㄱ-ㅎ|ㅏ-ㅣ|가-힣|\s]\*

위 정규 표현식에 일치할 경우 순수 한글로 된 책 이름으로 판단했다.

Exploratory Data Analysis(EDA)

다음은 분류기호 및 모델 언어별로 분류된 책들의 수를 나타낸 표이다.

| **분류기호/언어** | **한글 모델** | **다국어 모델** |
| --- | --- | --- |
| 0 | 9521 | 15066 |
| 1 | 12937 | 11604 |
| 2 | 4959 | 4107 |
| 3 | 39366 | 53631 |
| 4 | 4444 | 8104 |
| 5 | 7518 | 22207 |
| 6 | 14817 | 27650 |
| 7 | 15311 | 18911 |
| 8 | 11040 | 10698 |
| 9 | 47241 | 32935 |
| 10 | 13774 | 20985 |
| (계) | 180928 | 225898 |

프로젝트의 의의를 확인해보기 위해, 분류명이 이름에 직접 들어가 있는 책을 해당 분류명에 분류하는 간단한 가상의 분류기를 만들어 그 정확도를 계산해보았다. 계산 결과는 다음 표와 같다.

| **분류명** | **분류명포함책이름** | **실제분류일치** | **정확도 (%)** |
| --- | --- | --- | --- |
| 총류 | 0 | 0 | 0 |
| 철학 | 2331 | 1438 | 61.69 |
| 종교 | 437 | 287 | 65.68 |
| 사회학 | 531 | 441 | 83.05 |
| 언어 | 1354 | 463 | 34.19 |
| 자연과학 | 67 | 38 | 56.72 |
| 기술과학 | 2 | 0 | 0 |
| 경영학 | 327 | 282 | 86.24 |
| 예술 | 1211 | 681 | 56.23 |
| 문학 | 3192 | 2291 | 71.77 |
| 역사 | 4060 | 1704 | 41.97 |
| (계) | 13512 | 7625 | 56.43 |

위 결과를 통해 분류명 자체가 제목에 포함되어 있는 것만으로는 완벽하게 장르를 분류해낼 수 없음을 알 수 있다. 결국 책 이름으로 책을 분류하기 위해서는 단순 키워드만을 볼 것이 아니라, 제목의 문맥을 전부 고려해야만 의미 있는 결과가 나올 수 있음을 알 수 있었다.

아래는 분류기호 별 크롤링에 성공한 목차의 수를 나타낸 표이다.

| **분류기호** | **목차 수** |
| --- | --- |
| 0 | 5227 |
| 1 | 5512 |
| 2 | 1444 |
| 3 | 16104 |
| 4 | 2162 |
| 5 | 5113 |
| 6 | 7304 |
| 7 | 6982 |
| 8 | 4231 |
| 9 | 12101 |
| 10 | 5298 |
| (계) | 71478 |

**Method**

Model Description

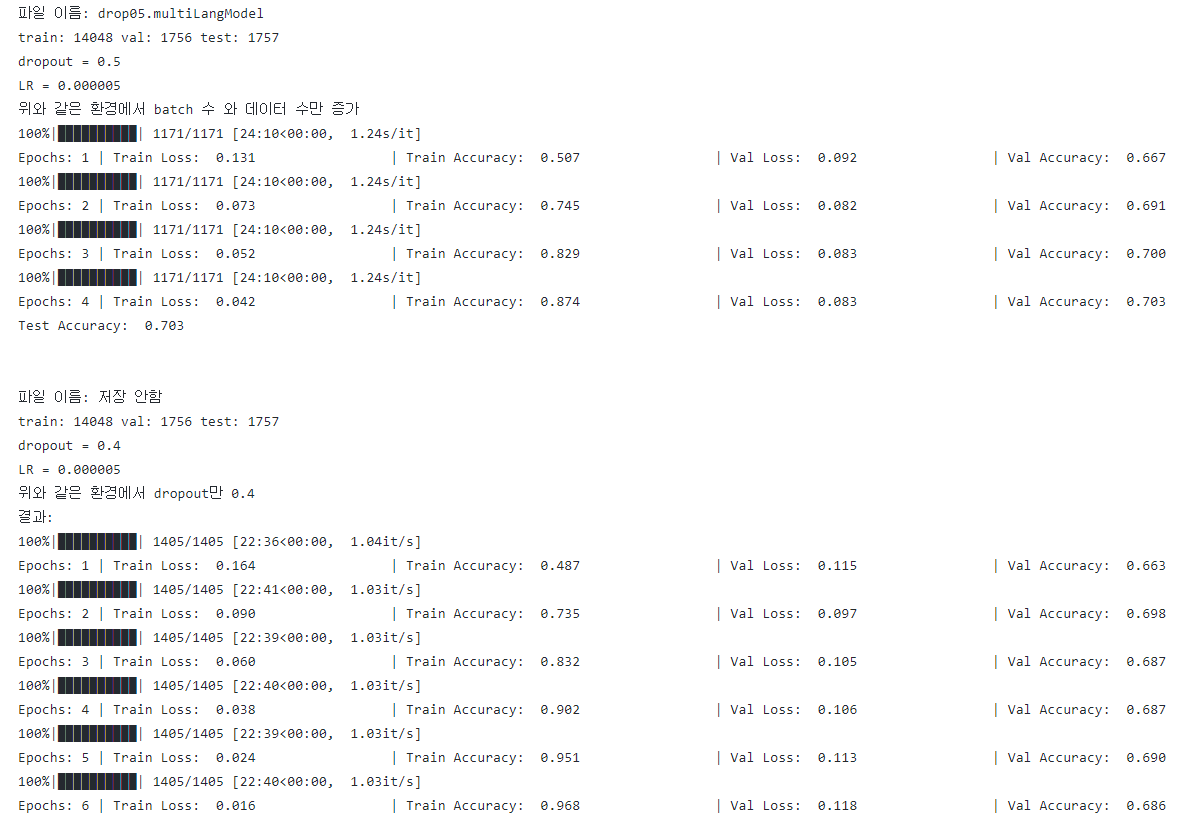
우리는 다양한 자연어 처리 분야에서 가장 좋은 성능을 내고 있는 BERT를 모델로서 사용하기로 했다.

구글의 Devlin(2018)이 제안한 BERT(Bidirectional Encoder Representations from Transformers)는 대용량의 데이터를 비지도 학습하고, 이를 토대로 언어 모델을 위한 신경망을 추가하는 사전 학습된 전이 학습 모델이다.

BERT는 대량의 데이터를 빠른시간에 학습하기 위해 링 리듀스(ring-reduce) 기반 분산 학습 기술을 사용하여, 십억 개 이상의 문장을 다수의 머신에서 빠르게 학습한다. 더불어, 파이토치(PyTorch), 텐서플루(TensorFlow), ONNX, MXNet을 포함한 다양한 딥러닝 API를 지원함으로써, 많은 분야에서 언어 이해 서비스 확산에 기여하고 있다.

그러나 BERT 모델에는 한국어 성능에 한계가 있어, 한국어 데이터에 대해서는 SKTBrain에서 공개한 Korean BERT pre-trained cased (KoBERT)를 사용하였다. (<https://github.com/SKTBrain/KoBERT>)

KoBERT는 기존 BERT의 한국어 성능 한계를 극복하기 위해 개발되었다. 위키피디아나 뉴스 등에서 수집한 수백만 개의 한국어 문장으로 이루어진 대규모 말뭉치(corpus)들을 학습하였으며, 한국어의 불규칙한 언어 변화의 특성을 반영하기 위해 데이터 기반 토큰화(Tokenization) 기법을 적용하여 기존 대비 27%의 토큰만으로 2.6% 이상의 성능 향상을 이끌어 냈다.

Setting Hyperparameters

프로젝트 중 이루어진 모든 학습들은 위와 같은 양식으로 Github에 기록하며 진행했다.

튜닝한 하이퍼파라미터들에는 다음 값들이 있다.

max\_len = 64

batch\_size = 64

warmup\_ratio = 0.1

num\_epochs = 5

max\_grad\_norm = 1

log\_interval = 200

frac = 1

test\_size = 0.2

learning\_rate = 5e-5

dr\_rate = 0.7

주요 하이퍼파라미터별 튜닝 과정은 다음과 같다. (동일한 데이터가 주어졌을 때를 전제로 한다)

Learning Rate의 경우, Validation Loss가 올라가는 시점을 Overfitting 시점이라고 보고 학습이 끝나는 시점이 최소 Val Loss가 될 수 있도록 비슷한 방식으로 최선의 값을 찾아갔다.

optimizer로는 높은 안정성의 AdamW를 사용하였다. weight\_decay 값을 적절하게 조정하였으며, scheduler는 cosine\_with\_warmup을 사용했다.

dropout rate 값은 커질수록 성능이 좋아지다 0.7일 때가 가장 좋았으며, 0.8부터는 다시 떨어지기 시작했다. 0.7까지는 overfitting을 방지하며 Val Loss를 최소화시킬 수 있었지만, 그 이상부터는 의미가 없는것으로 보인다

**Experimental Result**

Evaluate and Validate the Model

성능은 대체적으로 만족스러운 편이었다. 특히, 학습 데이터로 제목만 사용하였음에도 불구하고 75~79%정도의 정확도를 보였다. 아래는 koBERT 모델이 예측한 상위 세 개의 중 정답이 있을 경우를 계산한 결과인데, 정확도가 무려 96.5%인 것을 확인할 수 있다. (데이터는 한국어 제목 책들 중 20%를 무작위 추출한 것이다.)

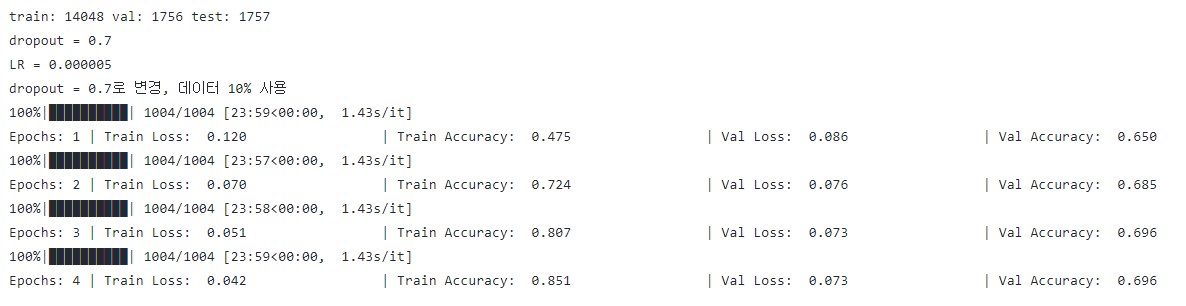


목차의 경우 제목보다 같은 항목 개수 대비 더 높은 정확도를 보이긴 했지만, colab의 사용시간 한계로 인해 그 이상 정확도를 끌어올리지 못하였다. 다만 제목의 경우 아무리 학습시켜도 80%가 한계일 것으로 보이는 반면, 목차의 경우 충분한 수의 데이터로 학습시키기만 한다면 훨씬 더 높은 성능을 보일 수 있을 것으로 기대된다.

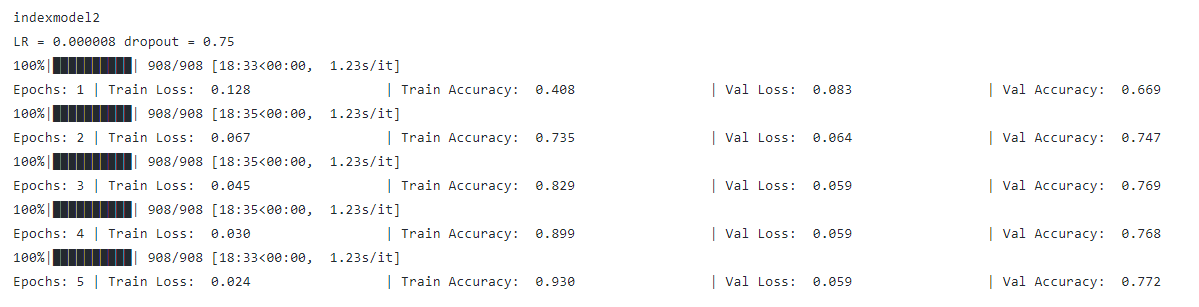
Novel or Unique Findings, Patterns and Insights

‘History’와 같은 1개의 단어로 이루어진 제목은 95% 확률로 역사로 예측하는 반면, 의학적 내용을 다루고 있는 ‘The History of Pain’ 같은 제목에는 기술과학과 자연과학 중 한가지로 확신하고 역사는 거의 고려하지도 않는걸 보아, 문맥을 확실하게 고려하여 예측하는 것을 확인할 수 있다.

또한, 제목 모델 이외에도, 목차 모델을 활용하여 자동화 분류의 가능성을 검토해보았다.

****

위의 내용은 14048개의 training 데이터를 활용한 다국어 모델의 성능이다

****

위의 내용은 목차를 활용하여 12707개의 training 데이터를 활용한 다국어 목차 모델의 성능이다. 더 적은 수의 데이터로도 더 우수한 결과가 나오는것을 확인할 수 있다. 알라딘 측의 크롤링 제한에 의해 더 많은 데이터를 확보하는데에는 어려움이 있었지만, 그럼에도 목차를 활용하는 분류의 가능성을 충분히 보여줄 수 있는 성과는 볼 수 있었다.

**Conclusion**

Project Summarization

먼저 크롤링을 통해 학교 도서관 데이터를 수집했고, 잘못된 데이터를 처리하고 청구번호를 추출하였다. 정규표현식을 사용하여 순수 한글 제목과 그렇지 않은 제목을 나누어 koBERT모델과 다국어 BERT 모델을 통해 학습시켰다. 주로 Learning Rate와 Dropout Rate에 초점을 맞추어 모델을 튜닝하였다. 모델들은 기본적으로 75% ~ 79%의 정확도를 보였으며, koBERT 모델이 예측한 상위 세 개의 중 정답이 있을 경우를 계산하면 정확도가 무려 96.5%에 달했다.

Main Advantages

제목에 의한 분류 정확도가 생각보다 높은것을 확인할 수 있었다. 새로운 책을 분류하는데 도움을 줄 수 있을 뿐만 아니라, 새롭게 책의 제목을 정하려 하는 사람에게도 책의 제목이 얼마나 직관적으로 책의 내용을 대변하고 있는지 확인하는 목적으로도 큰 도움을 줄 수 있을것이다.

Project Limitations

Colab의 GPU 성능 한계로 인해 장기간 학습을 진행하는데에는 무리가 있었다. 특히 목차 학습의 경우 각 항목마다 사용되는 단어의 수가 매우 컸기 떄문에 많은 학습을 진행하는 데에는 무리가 있었다. 거기에 알라딘에서 목차를 크롤링하는 과정에서, 하루에 5000개의 쿼리만 가능하다는 제한이 있어서 더더욱 데이터 수집에 큰 지장이 있었다. 결국 제한적인 데이터와 gpu 로 인해 목차 모델은 상대적으로 저조한 성능이 나오게 되었다.

특정 장르에 대한 책들이 부족하다는 점도 학습에 악영향이 있었다. 특히 목차가 있는 책들의 경우 사회학과 문학에 관한 책들은 만권을 넘는 반면, 종교에 관한 책은 1500권 가량에 그쳤다.

모델 관련 코드 및 모은 자료는 깃헙에 전부 업로드 했으며, 다음 링크를 통해 확인할 수 있습니다.

<https://github.com/diqnfl777/2022F-Ajou-ML-TEAM3>