도서를 통한 트랜드 파악 및 사용자 맞춤 도서 추천

(Identify trends through books and recommend customized books)

윤 경 서†
†국민대학교 AI빅데이터융합경영학과
(Kyungseo Yoon)
(†Kookmin University)

Abstract : 본 프로젝트는 현재 각 카테고리별 베스트셀러를 통해 트렌드 파악 및 사용자 맞춤형 도서 추천을 위해 교보문고에서 카테고리별로 도서 정보를 크롤링하여 데이터를 수집한다. 대용량 데이터인 Wikipedia 데이터를 이용하여 Word Embedding Model을 구축한다. 이를 활용하여 수집한 도서 데이터의 내용 벡터를 추출한다. 그 후, 사용자가 입력한 문장 및 단어와 도서 내용 벡터와의 유사도를 계산하여 도서들을 추천해 주는 기능을 제공한다. 또한, WordCloud, Keyword visualization, LDA 주제분석 등으로 현재 카테고리별 핵심 단어들을 보며 현시점의 트렌드를 파악한다. 이와 같은 도서 데이터 분석으로 현시점의 트렌드를 파악하고, 사용자 맞춤형 도서 정보 제공 및 추천을 진행하는 프로젝트를 제안한다.

Ⅰ.연구 목적 및 필요성

본 프로젝트의 목적은 교보문고의 도서 데이터를 활용하여 카테고리별 트렌드를 파악하고 사용자에게 맞춤형 도서 추천을 제공하는 것이다. 이를 위해 도서 내용의 특징을 추출하여 카테고리별 성격을 파악하고, 사용자의 취향과 일치하는 도서를 추천을 진행한다.

도서는 문화적인 요소로서 여전히 우리 생활에서 큰 역할을 하고 있으나, 독자들의 수는 점점 줄고 있다. 이러한 상황에서 독자들이 도서를 더욱 많이 구매하고 독서를 장려하는 것은 매우 중요한 과제이다. 독자들의 니즈가 다양화되고 사용자 맞춤형 서비스에 대한 수요가 증가함에 따라, 효율적이고 편리한 방법으로 독자의 취향을 반영한 맞춤형 도서 정보를 제공하는 시스템이 필요성이 제기되고 있다.

Ⅱ. 연구 배경 및 관련 연구

최근 관련 연구로는 도서 추천 시스템, 텍스트 마이닝, 자연어 처리 등이 있다. 이러한 연구들의 기술 발전과 함께 도서 추천 시스템은 독자들이 더 많은 도서를 발견하고, 개인 맞춤형 독서 경험을 제공함으로써, 출판 산업에 기여하고 있다. 과거의 도서 추천 시스템은 독자가 검색한 정보를 바탕으로 추천을 제공하는 방식이 대부분이었다. 그러나 여전히 독자의 이전 구매 이력이 없거나, 리뷰 혹은 평점 등의 사용자의 활동이 없는 경우에는 적용할 수 없다는 단점이 존재한다. 이와 같이 기존 연구들은 독자의 개별 취향과 니즈를 고려함에 있어 한계가 있었다.

따라서 독자들의 도서 선택과 만족도 향상을 위해 본 프로젝트를 진행했다. 독자들이 현재 도서 시장의 트렌드로 본인의 니즈와 적합한 단어 혹은 카테고리를 파악하고, 독서 후 사람들이 느끼는 만족감을 극대화할 수 있는 방향으로 추천 방향을 설정했다. 교보문고의 실시간 베스트셀러 리뷰를 분석한 결과, 도서 내용에 대한 리뷰가 대부분인 것을 알 수 있었다. 이를 통해 사람들이 독서 후 큰 만족감을 느끼는 부분은 도서의 내용이라는 것을 확인하여 이번 프로젝트에서는 도서 내용을 활용하고자 했다.

도서는 특정 시대의 상황과 흐름을 반영하는 중요한 자료로, 도서 카테고리별 분석은 현재의 트렌드를 파악하는 것이라 할 수 있다. 현재 트렌드가 되는 키워드들을 파악함으로써 도서 시장을 파악하고, 이를 통해 독자들에게 현시점에서의 카테고리별 정보를 제공하는 방법을 제안한다.

더불어, 독자가 더욱 개인화된 정보를 제공받을 수 있도록 도서의 내용을 분석하여 독자가 직접 입력한 단어 혹은 문장과 유사한 내용을 갖는 도서들을 추천하는 방법을 제안한다.

Ⅲ. 방법론

1) 데이터셋 구축

본 프로젝트에서는 교보문고 웹사이트에서 각 카테고리별로 도서 데이터를 크롤링하여 수집하였다. 수집한 데이터는 도서 제목, 저자, 카테고리, 도서 내용 정보를 포함한다. 교보문고 웹사이트의 '국내도서' 항목에서 '소설', '시/에세이', '인문' ,'가정/육아', '요리', '건강', '취미/실용/스포츠', '경제/경영', '자기계발', '정치/사회', '역사/문화', '종교', '예술/대중문화', '기술/공학', '외국어', '과학', '여행', '컴퓨터/IT', '잡지',' 청소년', 이렇게 총 20개의 카테고리를 선정하였다. 각 카테고리별 240권씩 총 4800권에 대한 데이터를 Selenium을 사용하여 데이터셋을 구축하였다.

2) Word Embedding Model 및 도서 내용 벡터 생성

Word Embedding Mode인 Word2Vec Model과 FastText Model을 구축하기 위해 앞서 수집한 데이터셋을 사용하고자 했다. 4800권에 대한 형태소 분석을 진행했을 때, 형태소 분석기별 어절 수와 불용어 처리 후의 어절 수, 형태소 분석 소요시간은 표 1과 같다.

표 1. 형태소 분석기별 어절 수 및 소요시간
Table 1. Number of words and required time for each morpheme analyzer

	klt2023	Okt	Stanza
어절 수	453,196	547,472	302,814
불용어 처리 후 어절 수	381,862	442,139	302,814
소요 시간	0:00:31.88	0:01:33.28	0:22:29.18

표 1에서의 어절 수를 보면, 전반적으로 Word Embedding Model을 구축하기에 너무 적다고 판단되어 Wikipedia 데이터를 사용하여 Word Embedding Model을 구축했다. Wikipedia 데이터를 KoNLTK로 형태소 추출을 진행한 ko_wiki_KMA_utf8.txt 데이터를 이용하여 불용어 처리와 전처리를 진행해 주었다. 구글링을 통해 한국어 불용어 리스트를 구축하여 불용어 처리에 사용하였다. 전처리로는 re 라이브러리의 정규표현식을 사용하여 HTML 태그, 구두점, 숫자, 영어와 한글 외의 문자 등을 제거해 주었다. 해당 데이터로 Word2Vec Model과 FastText Model을 구축했다.

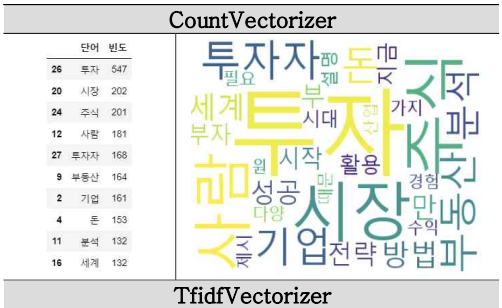
도서 내용 벡터를 만들기 위해 도서 내용에 해당하는 부분을 klt2023 형태소 분석기를 사용하여 명사만 추출하고, 불용어 처리를 해준 뒤, 하나의 도서 내용에 같은 명사가 여러 번 나오지 않도록 하기 위해 중복 제거를 해준다. 앞서 Wikipedia 데이터로 만든 FastText Model을 사용하여 각 도서의 내용에서 추출한 명사들의 단어 벡터를 구한다. FastText는 학습 데이터에 등장하지 않은 단어라도 n-gram 정보를 활용하여 단어 벡터를 생성할 수 있기 때문에 FastText Model을 사용하였다. 그 후 단어 벡터들의 합으로 도서 내용 벡터를 생성한다. 해당 방법을 통해 각 도서의 내용을 하나의 벡터로 표현한다. 도서 내용 벡터는 각 도서별 내용의 특징을 나타내는 중요한 정보를 담고 있으며, 이를 바탕으로 도서 정보 제공 및 추천에 활용된다.

3) 도서 카테고리별 WordCloud

우선 WordCloud로 각 도서 카테고리별 트렌드를 파악해 보고자 했다. WordCloud는 각 카테고리별 도서 내용을 가지고 진행했으며, 형태소 분석기는 stanza를 사용했다. 표 1을 보면 stanza의 경우 불용어 처리를 하기 전과 후의 명사 어절 수가 같다. 이는 가장 보수적으로 명사를 추출해 낸다는 것을 알 수 있다. 많은 수의 명사를 추출해 내지는 못하지만 다른 형태소 분석기에 비해 확실하게 명사로 판단되는 것들만 추출해 낸다고 판단하여 stanza를 사용하였다. WordCloud는 CountVectorizer와 TfidfVectorizer 두 개의 Vectorizer를 사용하여 생성했다. 아래의 표 2는 경영/경제 카테고리의 Vectorizer별 단어 빈도표와 WordCloud를 예시로 보여준다. 아래의 그림 1은 CountVectorizer로 만든 WordCloud의 예시이며, 그림 2는 TfidfVectorizer로 만든 WordCloud의 예시이다.

표 2. 경제/경영 카테고리 Vectorizer별 WordCloud
Table 2. WordCloud by Vectorizer in Economics/Business Category

CountVectorizer



TfidfVectorizer

그림 1. CountVectorizer WordCloud 예시

Fig 1. Example of CountVectorizer WordCloud

그림 2. TfidfVectorizer WordCloud 예시

Fig 1. Example of TfidfVectorizer WordCloud

생성된 WordCloud를 보면 각 카테고리별 트렌드를 살펴볼 수 있다. 표 2의 경제/경영 카테고리 같은 경우 투자, 주식, 기업 등과 같은 주식 관련 내용이 트렌드인 것을 알 수 있으며, 그림 2의 역사/문화 카테고리의 경우, 세계, 일본, 조선 등 일본 관련 내용이 트렌드인 것을 알 수 있다.

4) 도서 카테고리별 Keyword Visualization

다음은 keyword visualization으로 카테고리별 핵심 키워드를 살펴보고자 했다. 우선 textrank_KLT2023.py의 get_text 함수를 csv 파일을 받아올 수 있도록 수정하였고, get_nouns 함수에서 명사를 추출해낼 때에 불용어 처리를 하도록 수정해 주었다. 시각화는 plot4kcc_keyword.py에서 get_keywords, setEdges_keywords, visualization 함수를 사용하여 시각화하였다. WordCloud의 경우와 같이 각 카테고리별 키워드를 확인해 보기 위해 각 카테고리별로 시각화를 진행하였다.

그림 3. 시/에세이, 과학 카테고리의 Keyword Visualization

Fig 3. Keyword Visualization in Poetry/Essay, Science Category

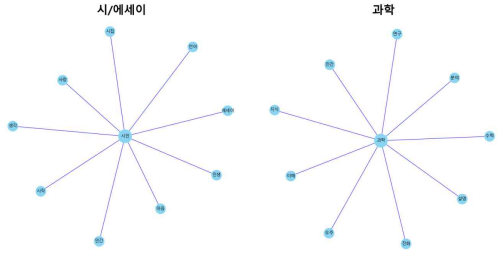


표 3. 일부 카테고리에 대한 Keyword Visualization 내용

Table 3. Keyword Visualization for Some Categories

역사/문화	자기계발	취미/실용/스포츠
역사	**인생**	**골프**
세계사, 인류, 중국, 일본, 조선, 전쟁	성공, 경험, 마음, 기업, 최고, 미국	선수, 설명, 축구, 방법, 이해, 사용

그림 3과 표 3을 살펴보면 각 카테고리의 특성을 잘 반영하면서 keyword를 추출해낸 것을 볼 수 있다. 역사/문화 카테고리의 경우 그림 2의 WordCloud 값과 비교해 보면, 세계사, 일본, 중국 등과 같은 단어로 세계사 중에서도 한국과 가까운 일본과 중국에 대한 내용이 많은 것을 알 수 있으며, 한국 역사에 대해서는 조선이라는 단어가 나오며 조선과 관련된 역사가 많이 수록되어 있다는 것을 알 수 있다. 역사/문화 카테고리 예시와 같이 Keyword Visualization으로 WordCloud에서 확인한 트렌드를 9~10개 정도의 키워드로 압축하여 카테고리의 특징을 파악할 수 있다.

5) Word Embedding 모델을 활용한 도서 추천

다음은 앞서 Wikipedia 데이터로 만든 Word Embedding Model과 도서 내용 벡터를 활용하여 도서를 추천을 진행한다. 우선 추천 과정에서 벡터 간의 코사인 유사도를 계산하는 코사인 유사도 함수를 정의해 주었다.

첫 번째 추천 서비스는 사용자가 입력한 문장으로 도서를 추천해 준다. 문장이 입력으로 들어오면 해당 문장의 명사를 추출하여 문장 벡터를 구하고, 해당 문장 벡터와 도서 내용 벡터들 간의 코사인 유사도를 구하여 코사인 유사도가 높은 상위 n개의 책을 추천한다. 명사를 벡터화하는 과정에서

Wikipedia 데이터로 만든 Word Embedding Model의 get_vector 함수를 활용하게 된다.

해당 서비스의 입력으로 '마음을 편안하게 해주는 위로의 글'이라는 문장을 주었을 때, Word2Vec model은 '행복을 담아줄게', '있는 그대로', '보이지 않는 곳에서 애쓰고 있는 너에게' 등의 도서를 추천해 주었고, FastText model의 경우, '당신이 좋아지면, 밤이 깊어지면', '나로서 충분히 괜찮은 사람', '당신과 아침에 싸우면 밤에는 입맞출 겁니다' 등의 도서를 추천해 주었다.

예시를 보면 각각의 Word Embedding Model 모두 입력된 문장과 관련 있는 도서들을 추천해 준 것을 볼 수 있다.

두 번째 추천 서비스는 단어들의 조합으로 도서를 추천해 준다. 사용자가 원하는 도서 내용을 positive word와 negative word로 표현하면, most_similar 함수를 사용하여 입력한 단어와 가장 유사한 단어들을 추출하고, 해당 단어들의 단어 벡터 합을 구하게 된다. 그리고 앞서 구한 단어 벡터들의 합과 도서 내용 벡터 간의 코사인 유사도를 구하여 코사인 유사도가 가장 높은 1권의 도서를 추천한다.

해당 서비스의 입력으로 positive word는 '모험', '전쟁', '성공'이며, negative word는 '공주', '연애'를 주었을 때, Word2Vec model은 '이미 시작된 전쟁'을, FastText model은 '로마인 이야기 2: 한니발 전쟁'을 추천해 주었다.

해당 서비스도 각각의 Word Embedding Model 모두 입력된 단어 조합과 연관이 있는 도서들을 추천해 준 것을 확인해 볼 수 있다.

세 번째 추천 서비스는 사용자가 입력한 단어로 카테고리와 도서를 추천해 준다. 단어 하나가 입력으로 들어오면 해당 단어의 단어 벡터와 각 카테고리명과의 코사인 유사도를 구하여 유사도가 가장 높은 카테고리를 선정한다. 그 후 해당 카테고리 내의 도서 내용 벡터와 사용자가 입력한 단어 벡터 간의 코사인 유사도를 구하여 유사도가 높은 상위 n개의 책을 추천한다.

해당 서비스의 입력으로 '모험'이라는 단어를 주었을 때, '소설'이라는 카테고리와 '해리 포터 시리즈 1~4권 세트', '해리 포터와 마법사의 돌 1', '해리 포터와 마법사의 돌 2' 등의 도서를 추천해 주었다.

마지막 서비스도 앞선 서비스들과 입력된 단어와 관련 있는 카테고리와 도서들을 추천해 준 것을 볼 수 있다.

6) LDA 분석

다음은 LAD 주제 분석이다. 본 단계에서는 앞에서 본 WordCloud, Keyword Visualization, 추천 서비스가 각 카테고리를 기준으로 진행되었기 때문에 해당 도서 데이터가 각 카테고리별 구분이 명확하게 되어 카테고리의 특징들을 잘 내포하고 있는지 확인한다.

우선 도서 내용에서 두 글자 이상인 단어를 추출한다. 이때, 최소 10개의 도서에서 등장하고, 전체 도서 수 중 70% 이상의 도서에서는 등장하지 않는 단어를 filtering 해주었다. 그 후, 단어별 빈도수를 구하여 corpus를 생성한다. 그리고 corpus 중 90%는 train, 10%는 valid로 나누어 LDA 모델 학습을 진행했다. LDA 모델의 topic 수는 100개로 설정했으며, loss는 log_perplexity로 혼란도를 측정하여 사용하였다. 가장 loss가 낮았던 -18.43의 혼란도를 갖는 LDA 모델을 사용하였다.

모델 구축 후, pyLDAvis를 통해 LDA 모델을 시각화하는 단계를 거친다. 이때, lambda 값은 keyword score의 weight와 관련된 것으로 0.4가 적당하다고 판단되어 0.4로 설정하였다. 그림 4는 pyLDAvis를 실행한 것이며, 그림 5는 일부 주제들에 대한 결과이다.

그림 4. pyLDAvis 결과
Fig 4. pyLDAvis Result

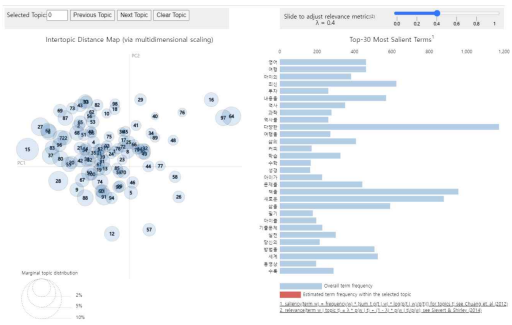


그림 5. pyLDAvis 일부 주제에 대한 결과값
Fig 5. pyLDAvis Results for some topics

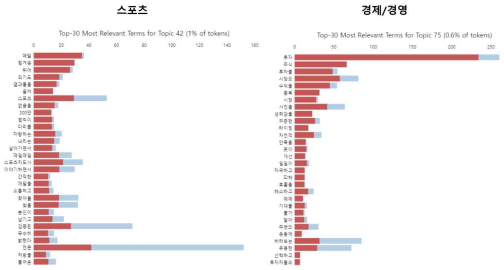


그림 4의 왼쪽에 출력되는 topic의 2차원 embedding vector는 다차원의 topic들을 차원 축소한 것이기 때문에 카테고리 간의 차이가 잘 보이지 않지만, 그림 5와 같이 각 주제를 선택해 보면, 카테고리의 특성들을 반영하고 있는 것을 알 수 있다. 그림 5의 47번 주제는 나온 단어를 보아 스포츠 카테고리일 것으로 예상되며, 75번째 주제는 경제/경영 카테고리일 것으로 예상된다. 100개의 모든 주제가 그림 5와 같이 명확한 주제를 보여주는 것은 아니지만 대부분의 주제가 카테고리를 예측할 수 있을 만큼 특징을 내포하고 있다는 것을 알 수 있다.

Ⅳ. 기대 효과

본 프로젝트에서는 교보문고에서 제공하는 도서 데이터를 활용하여, 카테고리별 도서 시장 파악 및 추천 서비스를 제공하였다.

자연어 처리 기술을 활용하여 도서 내용을 분석한 이번 프로젝트는 도서 추천의 효율성을 높일 수 있다. 즉, 독자가 원하는 도서에 대한 정보를 찾는데 드는 시간과 노력을 절약할 수 있다. 또한, 도서 데이터 분석을 통해 트렌드 및 키워드를 파악하고, 이를 독자들의 니즈에 적합하게 타겟팅을 하는 데에 도움을 줌으로써 도서 시장에서의 경쟁력을 향상시킬 수 있다.

도서 추천 서비스를 제공하는 플랫폼이나 서점에서 이번 프로젝트의 기술을 활용할 경우, 독자들의 독서 즐거움과 만족도가 높이고, 고객 서비스 품질을 향상시킬 수 있다. 추가적으로, 본 프로젝트의 확장 및 배포로 수집되는 다량의 도서 데이터와 고객 데이터는 도서 출판 및 도서 시장에서의 독자 니즈 파악, 마케팅 전략 수립 등에 활용될 수 있으며, 도서 산업의 발전에 기여할 수 있다.

따라서 본 연구는 현시점의 트렌드 파악 및 사용자 맞춤형 도서 추천 서비스를 제공함으로써, 독서 장려, 독자 만족도 향상, 도서 시장 경쟁력 강화 등의 다양한 효과를 기대할 수 있다.

References

[1] 남규현, 이현명, 강승식, “KoNLTK: 한국어 언어 처리 도구”, 제30화 한글 및 한국어 정보처리 학술대회, 2018

[2] 박은정, 조성준, “KoNLPy: 쉽고 간겨한 한국어 정보처리 파이썬 패키지”, 제26회 한글 및 한국어 정보처리 학술대회, 2014

[3] 원혜진, 이현영, 강승식, “대규모 텍스트 분석을 위한 한국어 형태소 분석기의 실행 성능 비교”, 한국정보과학회 2020 한국컴퓨터종합학술대회, 2020

[4] 정덕영, 이준석, 박상성, “워드클라우드를 이용한 기술 트렌드 분석”, 한국지능시스템학회, 2016

[5] 이대영, 이현숙, “LDA 토픽 모델링의 적정 토픽 수 결정 방법 탐색: 혼잡도와 조화평균법 활용을 중심으로”, 한국교육평가학회, 2021

부록 1

[1] 이철길, 한국어 불용어 리스트, “한국어 불용어”, https://deep.chulgil.me/hangugeo-bulyongeo-riseuteu/