비표준화 위해정보 통합분석 사업 연구보고서

권종익

2021.11.10

목차

# 개요

* 1. 초록
  2. 현상황 진단 및 문제 정의

# 본문

* 1. 레퍼런스 탐색
  2. 데이터 EDA
  3. 데이터 전처리
  4. 모델에 대한 설명
  5. 실험 설계

# 개요

* 1. 초록

1. 1차년도의 경우 SVM을 활용하여 사고 / 비사고 여부에 대한 이진분류 문제를 해결하였으나, 낮은 정확도( %)에 대한 이슈사항이 존재하였음
2. 해당 이슈를 해결하고자 기존 데이터에 대한 EDA를 실시한 결과, 1) 라벨링 오분류 사례 다수 존재 2) 심각한 라벨링 불균형 문제라는 문제점이 도출되었음
3. 해당 이슈를 해결하기 위해 1) 신규 데이터 수집 2) 라벨링 작업 재진행을 수행하였음
4. 재작업한 데이터셋을 기반으로 1) SVM, 2) KoBERT, 3)DHC(Deep Hierarchical Network) 세 종류의 모델에 대한 성능 평가를 진행하였으며, Specificity 기준 0% / 26% / 85% 의 성능을 기록하였음
   1. 현 상황 진단 및 문제정의
5. 1차년도 프로젝트의 경우 SVM 정확도가 다소 낮은 이슈사항이 있었음
6. SVM 정확도가 낮은 이유에 대하여 예상 가능한 원인을 탐색해본 결과 1) 데이터 차원 이슈 2) 모델링 차원 이슈 두 가지 원인으로 분류 가능하며, 각각의 범위에서 현 상황의 문제점을 EDA를 통해 진단하고 해결책을 도출하여 실제 서비스 수준에서 활용 가능한 모델을 도출하기 위한 연구 작업이 필요하다고 판단됨

# 본문

* 1. 레퍼런스 탐색

1. Deep Learning Based Text Classification : A Comprehensive Review(S Minaee et.al, 2020)

* Text Classification 아키텍쳐 중 현재 지배적인 위치를 점유하고 있는 아키텍쳐는 Pre-Trained Language Models(이하 PLM) 라고 주장
* PLM 아키텍쳐는 크게 Autoregressive 아키텍쳐(이하 AR 아키텍쳐)와 Autoencoding 아키텍쳐(이하 AE 아키텍쳐)로 분류 가능
* AR 아키텍쳐의 경우 직전의 토큰 예측 확률 분포를 토대로 다음 토큰의 확률을 예측하는 시계열적 구조를 채택하고 있음
* AE 아키텍쳐의 경우 토큰 예측값의 확률 변수들은 상호 독립적이라고 가정. 임의의 토큰을 마스킹(Masking) 처리하고 다른 예측값들의 예측 결과를 토대로 마스킹 토큰의 확률을 예측.

1. Deep Hierarchical Classfication for Category Prediction in E-commerce System(Dehong et al, 2020)

* 분류 문제 해결 시 직전 계층의 예측 결과가 다음 계층의 예측 결과에 영향을 주는 경우를 계층적으로 모델링
* 예를 들어, 직전 계층의 분류 예측 결과가 ‘어류’ 였다면, 그 다음 계층 분류 예측 결과는 ‘사슴’이 아니라 ‘상어’가 나오도록 학습 유도 필요
* 계층적 모델링을 위해 두가지 전략을 모델링에 채용
* Hierarchical Embedding Network: 상위 계층의 출력 벡터(Hidden Representation)을 하위 계층의 출력 벡터와 Concatenate
* Hierarchical Loss Network : 하위 계층의 예측 출력 값과 상위 계층의 예측 출력 값이 계층적으로 불일치하면 해당 불일치를 모델이 학습 과정에서 수정하도록 손실 값을 발생

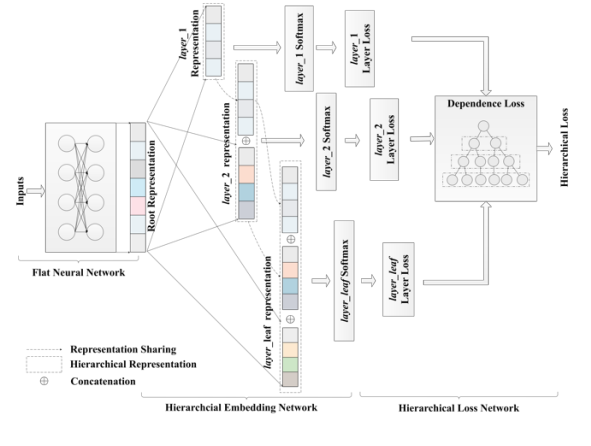


그림 1 - Deep Hierarchical Classification 얼개도

1. On the Variance of the Adaptive Learning Rate and beyond(Liu et al, 2020)

* 딥러닝 학습 분야에서 널리 사용되는 Optimizer인 Adam에 대해 분석하고, 약점을 도출한 후 해당 약점을 보완하는 보정항(Rectified Term)을 결합한 RectifiedAdam을 제안
* 수학적 증명은 다음과 같음
* gradient값이 N(0, Val)을 따른다고 가정할 때, 기존 ADAM의 파라미터인 learning\_rate의 분산은  
  역카이스퀘어 분포(l ~ scaled\_Inv X^2) 를 따른다고 가정할 수 있음

텍스트이(가) 표시된 사진

자동 생성된 설명

* 이 때, 이 분포의 분산은 시행 초기(t = 1)에는 무한대로 발산해서 불안정해진다고 주장

(아래 분산 추정식 참조)

텍스트이(가) 표시된 사진

자동 생성된 설명

* 위 분산 식을 해석하면, t가 무한대로 진행(t = ∞)됬다고 가정했을 때의 분산 기댓값이 가장 작아짐(즉, 안정적)
* 따라서, 시행이 무한대로 진행됬다고 가정했을때(t = ∞)의 분산 기댓값과 동일하도록 만드는 현재(t = now)의 분산 기댓값의 보정항을 도출(-> Rectified Term)

텍스트이(가) 표시된 사진

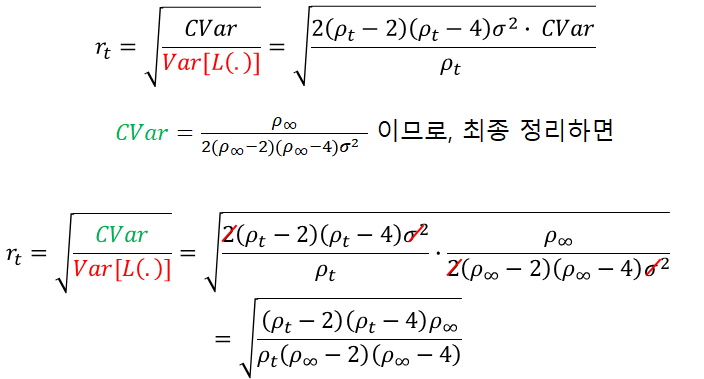
자동 생성된 설명

* 도출된 Rectifed Term을 ADAM항에 곱해 보정을 실시함.
* 이 때, (1)에서 도출한 Var[L(.)]은 계산하기가 복잡하고 안정적이지 않음.
* Var[L(.)]을 좀 더 안정화된 형태로 얻기 위하여, L^2(.) ~ scaled inv x^2에 대한 1차 테일러 근사식을 계산하면

텍스트이(가) 표시된 사진

자동 생성된 설명

* 따라서,



* 1. 데이터 EDA

1. 기존 데이터 EDA

* 뉴스 데이터에 대한 이진 분류가 엄밀하지 않음.
* 동일한 유형으로 분류 가능한 사안에 대해서도 한 쪽은 ‘비사고’, 한 쪽은 ‘사고’로 분류한 사례가 다수 존재

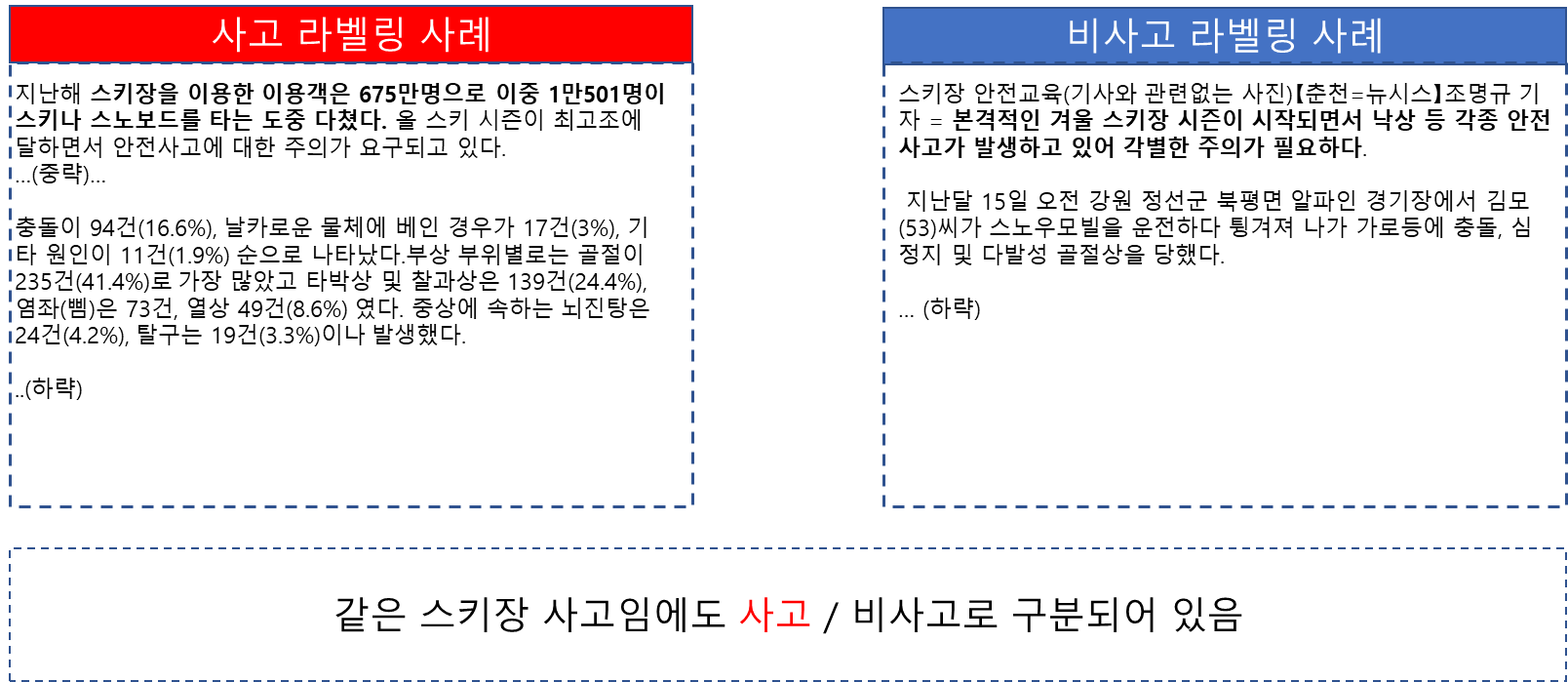
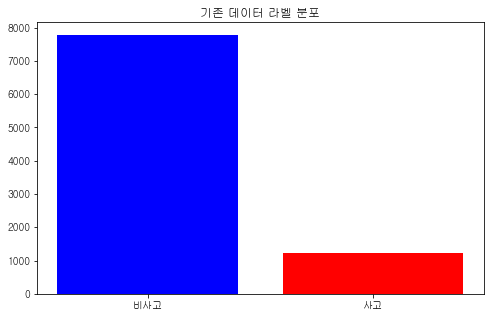


그림 2 - 기존 라벨링 데이터의 동일 유형의 다른 라벨링 사례 예시

* 해당 이슈를 해결하기 위해 다음과 같은 전략이 필요
* 정성적 유형 분류 : 뉴스 기사를 직접 확인하면서 뉴스 기사가 어떤 유형으로 분류 가능한지 정성적 / 직접적으로 확인
* 분류된 유형 기반 사고 이진 라벨링 : 분류된 뉴스 유형 기반으로 기사의 사고 / 비사고 여부를 판단하여 직접 라벨링
* 지나치게 큰 라벨링 불균형

|  |  |  |
| --- | --- | --- |
|  | **개수** | **비율** |
| **사고** | 7,779 | 86.35% |
| **비사고** | 1,230 | 13.65% |

표 1 - 라벨링 빈도표

|  |  |  |
| --- | --- | --- |
|  | **개수** | **상대 비율** |
| **사고** | 7,779 | 632.44% |
| **비사고** | 1,230 | 15.81% |

표 2 - 각 라벨에 대한 상대 비율

그림 3 - 기존 데이터 라벨링 분포

* 기존데이터의 경우 ‘사고’ 라벨링은 전체 데이터의 13%, ‘비사고’ 대비 16% 수준
* 라벨링 불균형을 판단할 수 있는 일반적인 기준은 없지만, 사고 내에서도 다양한 기사 유형이 존재함을 고려한다면 지금의 불균형은 모델 성능에 치명적인 요소로 작용 가능

1. 신규 데이터 EDA

* 학습용 데이터 신규 확보
* BigKinds API를 활용하여 수집한 2017, 2020년 데이터셋을 신규로 확보

|  |  |  |  |
| --- | --- | --- | --- |
|  | **데이터** | **생산년도** | **건수** |
| 1 | **BigKinds**  **수집 뉴스 데이터셋** | 2017 | 15,402 |
| 2 | 2020 | 15,671 |

표 3 - 신규 수집 데이터셋 현황

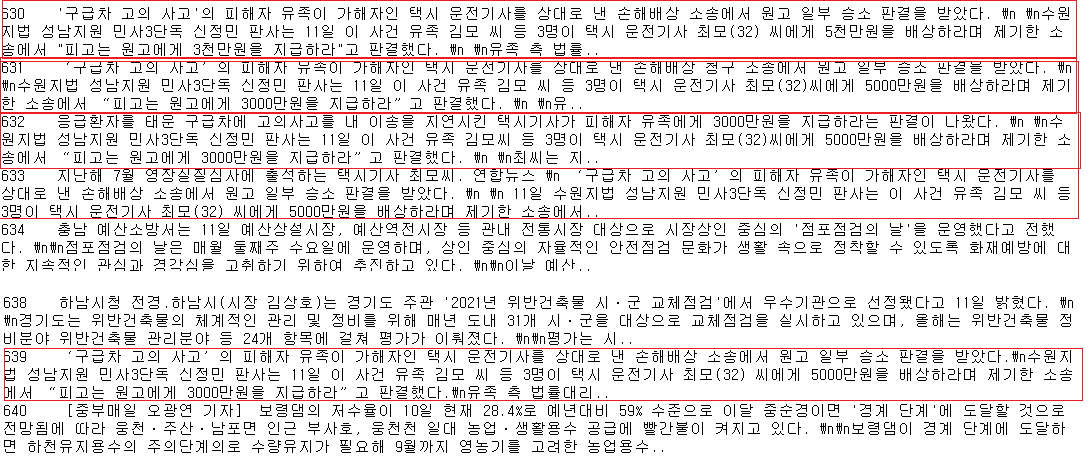
* 뉴스 데이터 문맥상 중복
* 완전히 동일하지는 않으나 동일한 주제를 다루고 있는 기사의 중복 수집
* 예를 들면, 아래와 같이 ‘구급차 고의 사고’에 대하여 동일한 주제와 동일한 문맥의 기사가 단어 순서만 조금 수정되어 다수의 언론사에서 생산되는 문제를 발견  
  

그림 4 - 문맥상 중복 데이터 예시

* 해당 문제를 해결하기 위하여 동일 문맥 기사라고 판단된 샘플에 대해서 자동으로 처리하는 알고리즘 개발 필요
  1. 데이터 전처리

1. 중복 데이터 제거

* <1.5-마>에서 제기된 중복 데이터 이슈를 처리하기 위해 ‘중복 데이터 판단 및 전처리 알고리즘’을 작성하여 전처리 자동화 수행

|  |
| --- |
| Ind\_memory(인덱스 정보 초기화)  Tfidf\_cosine\_self = 코사인 유사도(tfidf 행렬^T, tfdif 행렬)  Tfidf\_cosine\_triu = 상삼각행렬(tfidf\_Cosine\_self)  Threshold = 중복 문서 판단 Threshold  Error\_threshold = 인덱스 분산 기준 최종 판단 Thershold  For i in 뉴스 데이터셋:  Ind\_most\_simil = (Tfidf\_cosine\_triu[i] < threshold)  Ind\_memory\_new = ind\_memory[ind\_most\_simil] -> i번째 문서의 유사도 중 유사하지 않은 문서들만 저장  var = 분산[concat(i, ind\_memory\_new)]  If var < error\_threshold: -> duplicated 문서와 main 문서의 인덱스의 표준편차가 error\_threshold 이하  Ind\_memory = Ind\_memory[! Ind\_memory\_new] |

* 전처리 알고리즘 작동 결과는 다음과 같음

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **데이터** | **처리 전 건수** | **중복 판단** | **처리 후 건수** |
| 1 | 2017, 2020 BigKinds  수집 뉴스 데이터셋 | 31,703 | 630 | 31,074 |

1. 라벨링 작업 설계



그림 5 - 라벨링 작업 설계 절차도

* 라벨링 가이드라인 수립
* 알고리즘 이용 혹은 정성적 방법을 이용한 텍스트 유형 분류 실시
* 해당 단계에서 라벨링 작업의 기초이자 지침인 ‘라벨링 가이드라인’ 초안 도출
* 1차 사내테스트 수행
* 작업 측정 : 작업 난이도, 표준 작업 시간 측정
* 라벨링 가이드라인 적정성 확인 : 작업자별 라벨 일치도를 확인하여 작업자가 가이드라인에 기재되어 있는 기준에 따라 명확하게 작업이 가능한지 여부를 확인
* 기초 작업 설계
* 채용 계획 수립 : 측정된 작업 난이도, 표준 작업 시간 기반 선발 필요성 확인 및 채용 인원 결정
* 작업 관리 방법 결정 : 테스트 과정에서 도출된 예상 리스크를 관리하기 위한 관리 방법론을 설계
* 최종 사내 테스트 수행
* 선별절차 적정성 확인 : 도출한 문제 은행의 난이도 적정성, 변별력 여부 등을 확인

1. 라벨링 작업 준비

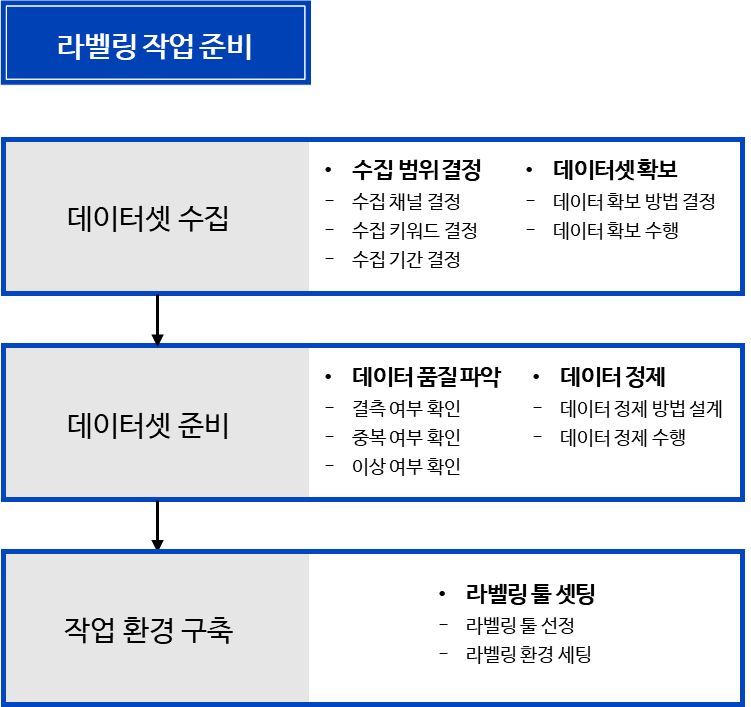


그림 6 - 데이터 작업 준비 프로세스

* 데이터셋 수집
* 수집 범위 결정 : 어떤 채널에서 수집할 것인지(ex. 빅카인즈 등), 그때 검색 키워드는 무엇으로 설정할 것인지( ex. 김치냉장고 등), 기간은 어떻게 설정할 것인지(ex. 2021-10-01 ~ 2021-10-21 등) 결정
* 데이터셋 확보 : 범위를 결정한 후 해당 범위의 데이터셋을 API를 통해 확보할 것인지, 수동 다운로드를 통해 직접 확보할 것인지 결정한 후, 데이터셋 확보 작업을 수행
* 데이터셋 준비
* 데이터 품질 파악 : 확보한 데이터에 결측값이나 명백한 이상(ex. 같은 단어 반복, 알 수 없는 키워드 나열 등), 중복값 여부를 확인할 수 있는 방법론 설계 및 확인
* 데이터 정제 : 확인된 품질 이슈를 자동으로 해결할 수 있는 방법론(ex. 유사도 측정 후 일정 유사도 이상 데이터셋은 제외하는 알고리즘 작성)을 설계하여 적용
* 작업 환경 구축
* 라벨링 툴 셋팅 : 라벨링 과업에 맞는 라벨링 툴을 선정하거나 자체 제작. 작업자가 바로 작업을 실시할 수 있도록 사전 환경 셋팅을 진행

1. 라벨링 작업 진행

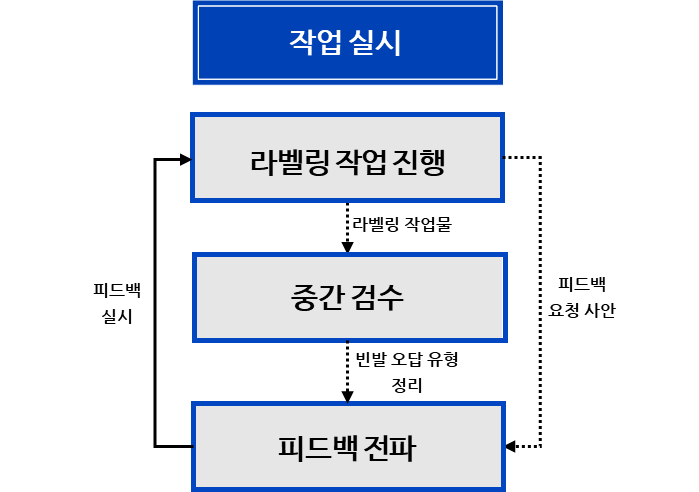


그림 7 – 작업 실시 절차도

* 다수의 라벨러가 동시에 작업을 진행하며 생길 수 밖에 없는 이해 수준의 차이, 해석 방향의 차이를 동일한 이해 수준 · 동일한 해석 방향으로 수렴시키는 것을 목적으로 함
* 라벨링 작업이 많이 진행된 상황에서는 라벨링을 다시 되돌리는 것이 사실상 불가능. 따라서 가능한한 초기에 이런 차이를 수렴하도록 해야함

1. 라벨링 작업 진행 결과

* 전처리 완료된 31,074건의 뉴스 데이터에 대하여 ㈜올빅뎃과 KOTITI시험연구원이 각각 50%, 50%씩 업무를 분담하여 라벨링 업무를 진행
* 9월 2일부터 10월 8일까지 약 한 달간 진행된 라벨링 업무를 진행하며 ㈜올빅뎃 측에서는 총 10회의 중간 피드백 회의를 진행하였으며, 10회의 라벨링 중간 검수를 진행하였다.
* 10월 8일 최종 라벨링 업무가 완료된 후, 라벨링 완료된 31,074건의 뉴스 데이터에 대하여 각 라벨러별 최종 샘플 검수를 통해 라벨링 품질을 검증하였으며, 최종적으로 활용 가능한 12,699건을 도출하여 학습용 세트로 최종 확정
  1. 모델에 대한 설명

1. Deep Hierarchical Network 차용

* 라벨링 완료된 10가지 뉴스 유형을 다중 분류하고, 다중 분류 결과를 토대로 사고 / 비사고 이진 분류를 수행하도록 설계
* 첫 번째 모델인 ‘문맥 파악 Model’은 KoBERT를 사용. 문맥 정보를 Hidden Representation으로 종합하여 분류 Model들에 전달
* 두 번째 모델인 ‘유형 분류 Model’은 KoBERT에서 산출된 Hidden Representation을 기반으로 해당 뉴스의 유형을 분류
* 세 번째 모델인 ‘사고 이진 Model’은 문맥 Model에서 전달받은 Hidden Representation과 유형 분류 Model의 Hidden Representation을 결합하여 사고 여부를 판별

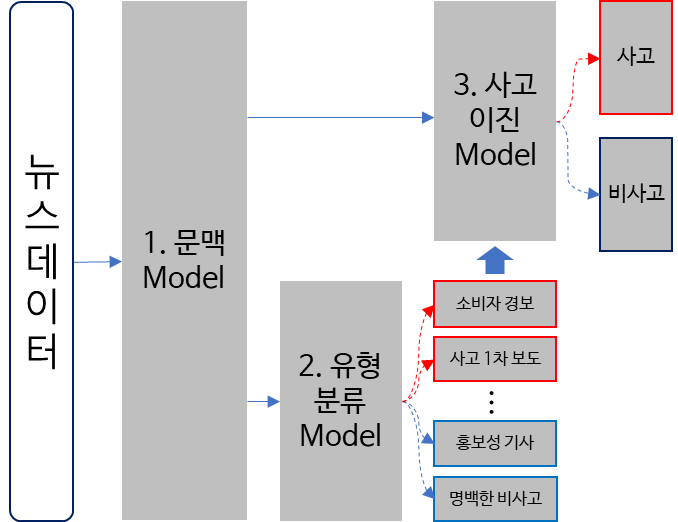


그림 8 - DHC 실제 적용 개념도

1. 모델 손실 함수 정의

* 논문(Dehong et al, 2020)에서 제안하는 계층적 손실 함수를 개량하여 계층 구조 위반에 대한 손실값을 각 계층의 이진 / 다중 분류 손실값을 차용하여 사용하도록 수정
* 논문 제안 손실 함수 형태 : 값을 고정된 상수 혹은 각 분류 Model의 손실값을 차용한 형태를 활용할 것을 권고
* 프로젝트 적용 손실 함수 형태 : 값을 각 분류 Model의 손실값을 차용한 형태로 변형
  1. 실험 설계

1. 벤치마크 모델 선정

* 1차년도 적용 모델인 SVM, 선정 후보 SOTA 모델인 KoBERT와 DHC를 선정

1. 실험 설계 방향

* KoBERT의 경우 특정 Epoch 도과 이후 성능이 급격히 떨어지는 이슈가 발생
* 이를 현 데이터셋에 대한 KoBERT 학습의 특성이라고 한다면, 이를 기준으로 실험 설계를 진행할 필요가 있음
* 사고 => 비사고 오분류가 비사고 => 사고 오분류보다 더 치명적인 이번 프로젝트 특성 상 ‘사고(라벨 0)’를 기준으로 분류 성능을 평가하는 지표인 ‘특이도(Specificity)’를 가장 높은 우선순위로 설정
* 그 외에 전통적으로 모델 분류 성능에 많이 쓰이는 평가 지표인 ‘정확도(Accuracy)’, ‘정밀도(Precision)’, ‘재현율(Recall)’, ‘F1-Score’를 부(附)평가지표로 선정
* 한편, 일반화된 성능을 평가하기 위해 8-Fold Validation을 수행
  1. 실험 결과

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **Specificity** | **F1\_score** |
| **SVM** | 79.24% | 79.24% | 100% | 0% | 88.42% |
| **KoBERT** | 80.02% | 84.57% | 93.91% | 26.48% | 87.98% |
| **DHC** | 94.37% | 95.63% | 97.11% | 85.23% | 96.36% |

* 1. 시각화 결과

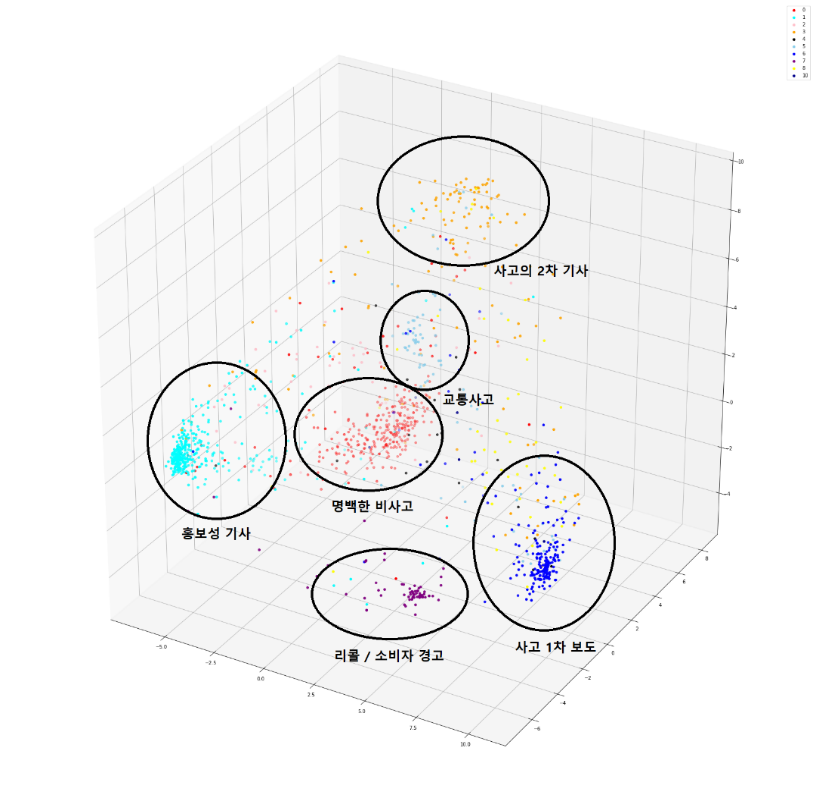


그림 9 - DHC 뉴스 다중 분류 시각화

1. 다중 분류 결과

* 11개 뉴스 유형 분류 중 6개 뉴스 유형은 상대적으로 군집이 잘 이루어진 것을 확인 가능

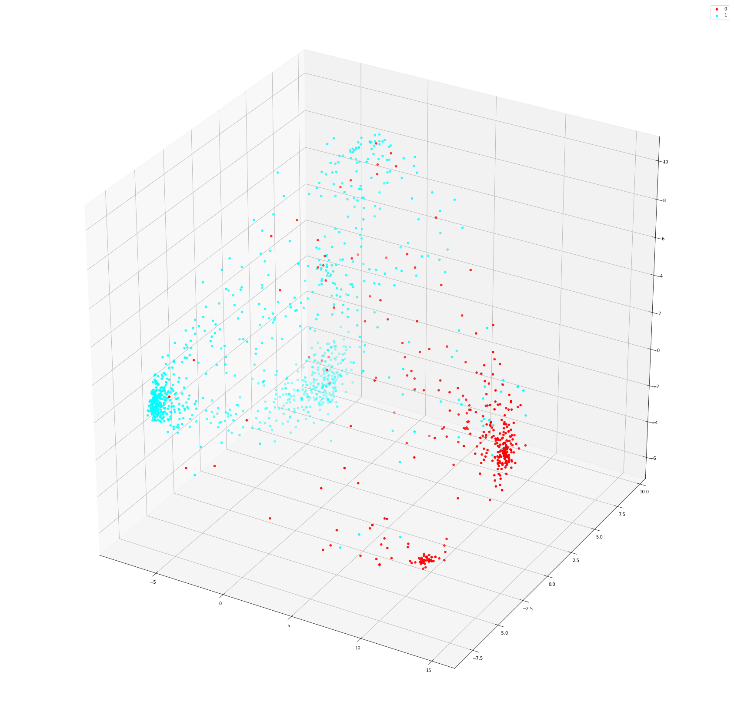


그림 10 - DHC 뉴스 이진 분류 시각화

1. 이진 분류 결과

* 뉴스 다중 분류 결과물을 바탕으로 사고 / 비사고 여부의 이진 분류 결과물을 시각화한 결과 사고 / 비사고가 비교적 잘 분류되 있는 상태를 확인 가능함

# 결론

기존 데이터셋에 대하여 이슈사항을 발견하고, 해당 이슈사항을 해결하기 위하여 1) 데이터 전처리 재실시 2) 신규 모델 적용을 수행하였다. 그 결과 SVM은 물론이고 기존 데이터셋으로 적합한 KoBERT 모델의 Specificity 58%를 뛰어넘는 85%의 Specificity를 기록하였다.

또한, 단순 이진분류 모델이 아닌 계층적 다중분류 모델인 DHC를 실제 적용함으로서 1) 텍스트 유형 우선 분류 2) 유형 분류 결과 기반 이진 분류라는 실제 인간이 뉴스에 대한 유형을 탐색할 때 수행하는 절차를 모델이 모사하게 만드는데 성공하였다.

라벨링을 최초 설계부터 데이터 산출까지 진행한 경험은 차년도 예정 업무인 ‘다중 분류’에 적용이 가능할 것으로 판단되고, 계층적 이진분류 모델인 DHC는 차년도 예정 업무인 ‘다중 분류’의 계층성(중분류 -> 소분류로 이어지는 계층 구조)을 조금의 변형으로 바로 반영할 수 있어 프로젝트 기간 단축에 도움이 될 것으로 생각된다.