




# [STA518 FINAL PROJECT]

EXTRACTIVE SUMMARIZATION

USING ML METHODS

2021-2 통계분석방법론

2021020357 이준희



# 목 차

## 제 1 장. 프로젝트 개요

1.1 추출 요약

1.2 프로젝트 설명

1.3 데이터 소개

1.4 탐색적 데이터 분석

1.5 모델 평가 지표

1.5.1 정확도

1.5.2 F1 점수

1.5.3 ROC-AUC

## 제 2 장. 데이터 분석

2.1 데이터 전처리

2.2 설명 변수

2.2.1 Mecab 형태소 분석기

2.2.2 TF-IDF 가중치

2.2.3 코사인 유사도

2.2.4 설명 변수 생성

## 2.3 모형 구축

2.3.1 모형 구축을 위한 데이터 변환

2.3.2 순열 변수 중요도 ( Permutation Feature Importance )

2.3.3 로지스틱 회귀 분석

2.3.4 의사 결정 나무

2.3.5 랜덤 포레스트

2.3.6 XGBOOST ( eXtreme Gradient BOOSTing )

## 제 3 장. 결론

3.1 분석 결과

3.2 추후 연구 방향

## 제 4 장. 부록

4.1 참고 문헌

## 제 1 장. 프로젝트 개요

---

### 1.1 추출 요약

텍스트 요약은 자연어처리(NLP, Natural Language Processing) 분야의 주요 연구 분야 중 하나로서, 언어를 이해하고 지식을 추출하여 새로운 가치를 창출하는 과정에 있어 중요한 역할을 한다. 최근 인터넷 매체의 발달로 다양한 콘텐츠를 소비할 수 있는 환경에 처한 콘텐츠 소비자들을 대상으로 AI를 이용한 요약기술은 핵심 내용을 신속하고 정확하게 파악하는 과정에 도움을 줌으로써 다양한 수요를 파생시킨다. 예컨대 복잡한 법률 문서를 요약해 핵심 내용만을 파악할 수 있도록 해주고, 뉴스 기사를 요약해 중요한 소식을 빠르게 접하는 등의 사례가 있을 것이다.

텍스트 요약은 크게 추상 요약(Abstractive Summarization)과 추출 요약(Extractive Summarization)으로 나눌 수 있다. 추상 요약이란 원문에 없던 문장이라도 핵심 문맥을 반영한 새로운 문장을 생성해서 원문을 요약하는 방법이다. 추상 요약을 통해서 원문에 존재하지 않는 새로운 단어로 구성된 문장을 만들어 낼 수 있고, 딥러닝의 Seq2seq(Sequantial to Sequantial) 방법이 주로 사용된다. 반면 추상 요약과 달리 추출 요약이란 원문에서 중요한 핵심

문장 또는 단어구를 몇 개 뽑아서 이들로 구성된 요약문을 만들게 되고, 추출 요약의 결과로 나온 요약문의 문장이나 단어들은 전부 원문에 있는 문장들로 구성되게 된다. 그렇기 때문에 추출 요약은 텍스트가 중요한 문장으로 사용이 되는지 아닌지에 대한 이진 분류로 볼 수 있을 것이다. 본 프로젝트에서는 이진 분류에 사용되는 다양한 지도 학습 모델을 사용하여 추출 요약에 접근하고자 한다.

## 1.2 프로젝트 설명

본 프로젝트에서는 [통계 분석 방법론]에서 학습한 지도 학습 모델들을 이용하여 추출요약 모델을 구축할 것이고,

[생명과학 연구를 위한 통계적 방법] 4장. 로짓 및 로그선형 분석

[응용데이터분석] 18장. 나무모형

을 참고할 것이다.

이진 분류에 많이 사용되는 통계적 모형인 로지스틱 회귀 분석(Logistic Regression)부터 시작해 불순도를 낮추는 방향으로 가지를 성장시키는 나무 모형(Decision Tree), 나무 모형을 앙상블하는 랜덤 포레스트(Random Forest) 모형, GBM(Gradient Boosting Model)을 발전시킨 XGBOOST(eXtreme Gradient BOOSTing) 모형을 사용하고자 한다.

각 모형의 작동 원리를 살펴보고 모형의 분류 성능을 정확도, F1 점수, AUC를 통해 살펴보고자 한다. 이 때, 각 모형의 엄밀한 비교를 위해 5-겹 교차 검증(5-Fold Cross Validation)을 통해 결과를 도출할 것이다.

### 1.3 데이터 소개

본 프로젝트를 위해서 텍스트, 텍스트 요약에 사용된 문장들이 담겨 있는 데이터를 구하고자 했고, 필자는 AI HUB 사이트의 개방 데이터 목록 중 한국지능정보사회진흥원이 구축한 문서 요약 텍스트 데이터를 사용한다.

( <https://aihub.or.kr/aidata/8054> )

원본 JSON 데이터 신문 기사를 파이썬의 데이터 프레임 형식으로 변환하여 약 30만 행의 데이터를 얻었으나 Computation cost를 고려해 신문 기사 개수가 가장 많은 상위 5개 신문사들의 데이터만을 층화 추출(Stratified Sampling)해 약 1만 행의 데이터로 축소하였다. 데이터에서 추출 요약에 필요한 변수만을 추출하면 아래와 같은 형태를 띈다.

[ 추출 요약에 사용할 가공 데이터 ]

media_name	article_original	abstractive	extractive
디지털타임스	[우리나라 수출이 지난달까지 9개월 연속 감소했다., 반도체, 석유화학 등 주력 수...	[산업통상자원부는 8월 수출이 전년 동기 같은 달보다 13.6% 줄어든 442억달러로 ...	[3, 4, 15]
아주경제	[KT는 현지시간으로 내달 6일부터 11일까지 독일 베를린에서 열리는 유럽 최대 가...	[3일, KT는 내달 6일부터 11일까지 독일 베를린에서 열리는 유럽 최대 가전 I...	[0, 4, 6]
매일경제	[한국의 11년 만에 세계청소년아구선수권대회(U-18 야구 월드컵) 우승에 비상등이...	[세계청소년아구선수권대회에 참가한 한국 청소년야구대표팀이 슈퍼라운드 첫 경기만에 대...	[0, 2, 12]
매일경제	[국제구호개발 NGO 굿네이버스(회장 양진옥)가 유산기부자 모임 '더네이버스레거시'를...	[국제구호개발 NGO 굿네이버스가 유산 기부를 원하는 사람들에게 법률, 금융 등 맞...	[0, 6, 7]
디지털타임스	[홍남기 경제부총리 겸 기획재정부 장관은 16일 '주택을 통한 불로소득은 어떠한 경...	[홍남기 경제부총리는 16일 '주택을 통한 불로소득은 어떠한 경우에도 절대 허용하지...	[0, 3, 10]

Media\_name : 신문사 이름

Article\_original : 신문기사 원본

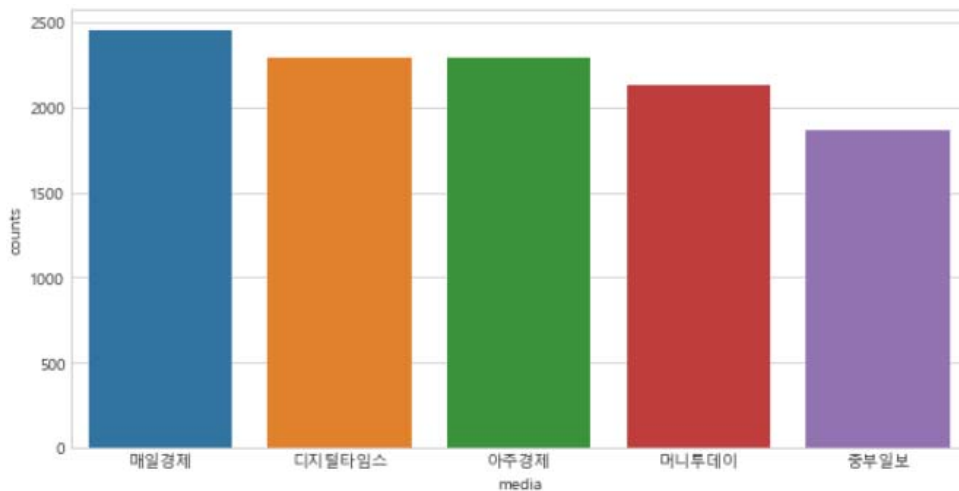
Abstractive : 사람이 직접 작성한 요약본

Extractive : 요약에 사용된 핵심 문장

#### 1.4 탐색적 데이터 분석 (EDA)

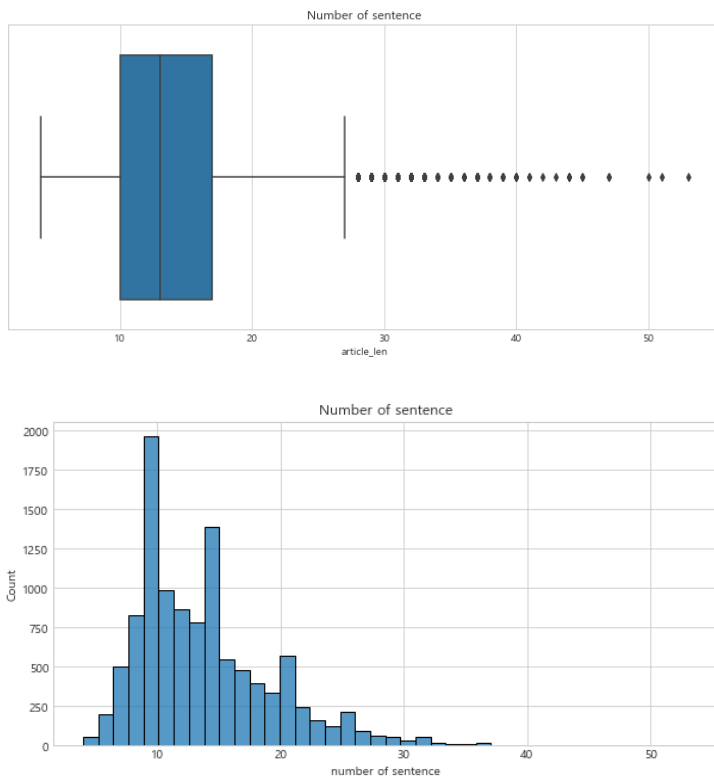
추출한 데이터에서 총 5개의 신문사가 있고, 각 신문사 데이터의 숫자는 약 2000개 정도로 동일한 것을 알 수 있다. 이는 앞에서 신문사 별로 층화 추출을 했기 때문이다.

[ 추출 요약 데이터 내의 각 신문사 데이터 히스토그램 ]



신문 기사에서 특수문자, 공백 등을 제거한 후 기사 내 문장들의 개수 분포를 살펴보았다. 최소 4개의 문장으로 구성된 신문 기사부터, 최대 53개의 문장으로 구성된 신문 기사가 있었으며 평균적으로 신문 기사는 13.9개의 문장을 가지고 있었다.

[ 기사 내 문장들의 개수 분포 시각화 그림 ]

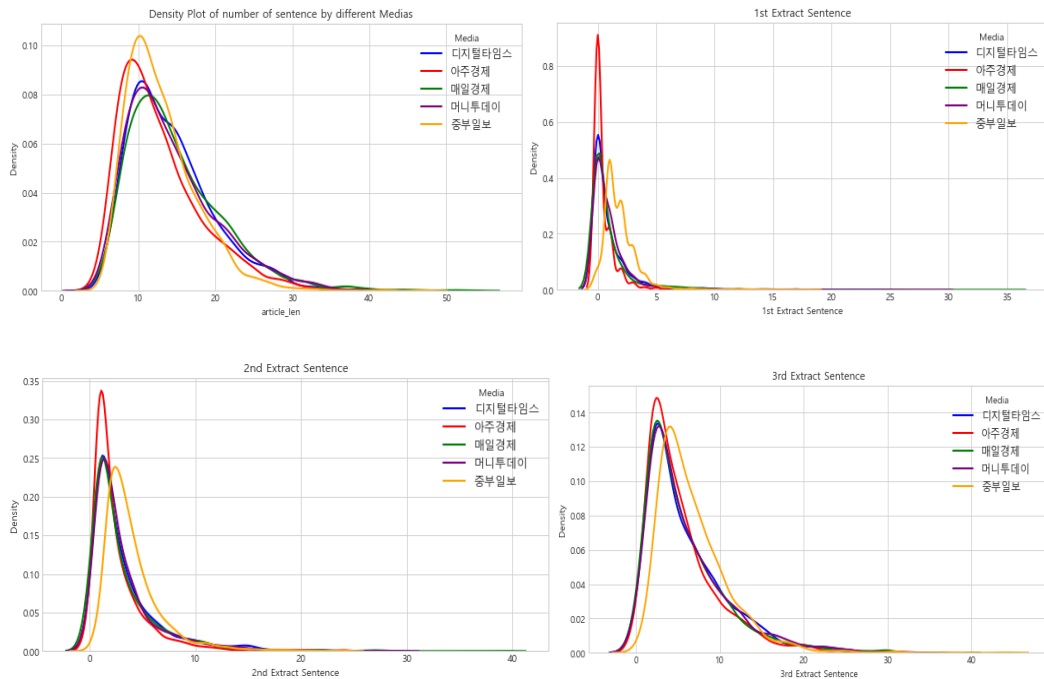


기사 내 문장들의 개수는 신문사 별로 큰 차이가 있지는 않았다. 하지만 신문사마다 요약에 사용이 된 문장의 위치는 상이했는데, 요약 문장 3개 중



에 위치 상 맨 앞에 있는 첫 번째 요약문장의 분포를 살펴보면 ‘아주경제’의 경우 80% 이상이 첫 번째 문장을 사용했고, ‘중부일보’의 경우 첫 번째부터 다섯 번째 문장까지 중 골고루 중요 문장이 분포되어 있었다. 두 번째 요약문장과 세 번째 요약문장의 경우에도 대다수가 기사 초반에 분포하고 있었으며, ‘아주경제’의 경우 조금 더 문장 앞쪽에 요약 문장이 있고 ‘중부일보’가 요약 문장이 골고루 퍼져 있는 경향성은 유지되고 있었다.

#### [ 기사 내 요약 문장 위치 분포 시각화 그림 ]



기사들에서 ‘지난’, ‘사업’, ‘기업’, ‘경기’ 등의 단어가 가장 많이 등장한다. 하지만 각 기사마다 다루는 소재가 다르기에 등장하는 단어들의 특징이 상이한 부분이 있는 것 또한 구름 그림(Wordcloud)을 통해 살펴볼 수 있다.

[ 신문사 별 신문 기사 내 단어의 구름 그림 ]



이와 같은 데이터 탐색을 통해 신문사마다의 요약 문장 위치의 차이와 등장 단어의 차이를 고려해 파생변수들을 만들면 모형이 해당 문장이 요약에 사용 되는지 아닌지에 대한 이진 분류를 수행하는 것에 도움이 될 수 있을 것이라는 사실을 알 수 있었다.

## 1.5 모델 평가 지표

### 1.5.1 정확도

본 프로젝트에서는 요약에 사용된 문장을 1, 요약에 사용되지 않은 문장을 0 으로 분류하게 된다. 구축한 모형의 예측 값과 실제 모형의 값을 비교하는 혼동 행렬(Confusion Matrix)를 구성할 수 있다. 혼동 행렬은 TP(True Positive), TN(True Negative), FP(False Positive), FN(False Negative) 네 개의 요소로 구성된다.

[ 혼동 행렬의 구성 요소 ]

		PREDICTIVE VALUES	
		POSITIVE (1)	NEGATIVE (0)
ACTUAL VALUES	POSITIVE (1)	TP	FN
	NEGATIVE (0)	FP	TN

각 요소에 대해 설명하자면

TP 는 실제 True 이고, 분류모델에서 예측이 True 라고 판단된 경우이다.

TN 는 실제 False 인데, 분류모델에서 예측이 False 라고 판단된 경우이다.

FP 는 실제 False 인데, 분류모델에서 예측이 True 라고 판단된 경우이다.

FN 는 실제 True 이고, 분류모델에서 예측이 False 라고 판단된 경우이다.

이러한 혼동 행렬로부터 정확도를 유도할 수 있으며 식은 다음과 같다.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

하지만, 정확도로 모델을 평가할 때는 데이터 불균형을 고려해 주어야 한다.

본 데이터는 불균형 데이터로 요약에 사용되지 않는 문장(0)이 약 80%의 비율을 차지한다. 모든 예측을 요약에 사용되지 않는 문장(0)으로 진행할 경우 약 80%의 정확도를 얻을 수 있다는 점을 감안했을 때 80% 이상의 정확도를 얻을 수 있어야 유의미한 모델이라고 할 수 있을 것이다. 또한 정확도만으로 모델을 평가하게 되면 요약에 사용되는 문장(1)과 요약에 사용되지 않는 문장(0) 각각의 경우를 얼마나 잘 예측하는지에 대한 정보를 제공하지는 않는다는 점에서 한계가 있다. 그래서 정확도 이외에도 평가 지표들을 이용해서 모델 평가를 진행하고자 한다.

### 1.5.2 F1 점수

F1 점수는 정밀도(Precision)과 재현율(recall)의 조화평균으로 정의된다.

정밀도는 모형이 참이라고 예측한 관측치의 수에서 실제로 참인 관측치의 수이고, 재현율은 실제로 참인 관측치의 수에서 모형이 참이라고 예측한 관측치의 수로 정의되고 수식은 아래와 같다.

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$\text{F1-score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

정밀도와 재현율은 Trade-Off 관계에 있다. 그렇기 때문에 모형 초모수 조정을 통해 좋은 정밀도와 재현율을 가지는 모형을 구축하는 것이 중요하며 F1 점수를 통해 이를 확인할 수 있다. 본 프로젝트에서는 불균형 데이터를 다루고, 신문 기사에서 핵심이 되는 문장(1)을 잘 분류하는 것이 중요하기 때문에 각 모형의 F1 점수를 살펴보고자 한다. F1 점수는 크게 Micro 방식과 Macro 방식으로 평가할 수 있다. Micro 방식이란 각 클래스(class)의 관측치를 고려하여 F1 점수를 가중평균으로 구하는 것이고, Macro 방식이란 각 클래스의 관측치를 고려하지 않는 방식이다. 본 프로젝트에서는 핵심이 아닌 문장(0)의 비율이 약 80% 정도를 차지하기 때문에 문장(1)의 분류 성능을 조금 더 반영할 수 있는 Macro 방식을 통해 모형들을 살펴보고자 한다.

### 1.5.3 ROC-AUC

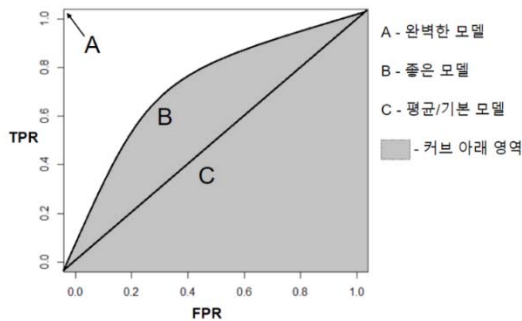
ROC curve는 이진 분류에서 많이 사용되는 개념으로 X축을 FPR, Y축을 TPR로 두어 모델이 양성을 예측했을 때, 얼마나 잘 맞추고 있는지를 나타내어 주는 그래프이다. FPR과 TPR의 식은 다음과 같다.

$$FPR = \frac{FP}{TN+FP}$$

$$TPR = \frac{TP}{TP+FN}$$

이 때, 양성(1)과 음성(0)을 판단하는 기준을 threshold라고 하는데, 주어진 데이터 내에서 threshold를 바꾸면서 FPR과 TPR의 다양한 조합을 계산하고 해당하는 조합들을 나타낸 그림을 ROC curve라고 한다.

[ ROC curve 그림 ]



이러한 ROC curve 곡선의 아래 영역을 ROC-AUC라고 정의한다. 좋은 모델일수록 TPR 값이 높아 실제 양성인 자료들을 양성이라고 잘 예측했을 것

이고, FPR 값이 낮아 실제 음성인 자료들을 양성이라고 예측하는 경우가 적을 것이다. 그렇기 때문에 ROC curve가 북서쪽으로 갈수록 좋은 모델인 것을 의미하게 되고, 이 때 ROC-AUC는 1에 가까워지게 된다.

## 제 2 장. 데이터 분석

---

### 2.1 데이터 전처리

분석에 필요한 텍스트 데이터를 만들기 위해 다음과 같은 과정을 수행한다.

1. 필수적인 특수 문자 이외의 특수 문자 제거
2. 2칸 이상의 공백을 제거

※ 여기서, 필수적인 특수 문자는 [+ @\$?W-!<&;()=/\_>"'%#`.~W]이다.

해당 과정을 통해 한글, 영어, 숫자만으로 구성된 텍스트 데이터를 구성한다.

### 2.2 설명 변수

전처리를 통해 핵심 텍스트만을 뽑아낸 이후에는 중요 문장 이진 분류 모델을 구축하기 위해서 설명 변수를 만들어 주었다. 설명 변수 생성 과정 설명에 앞서 변수 생성에 사용된 개념인 Mecab 형태소 분석기, TF-IDF 가중치, 코사인 유사도에 대한 개념을 설명하고자 한다.



### 2.2.1 Mecab 형태소 분석기

전통적인 형태소 분석기는 보통 규칙 기반(rule-based) 방법론으로 설계된다. 하지만 엄밀한 규칙은 유연한 언어를 해석하는 과정에서 많은 예외를 발생시키고, 여러 규칙들이 상반될 경우 규칙의 우선순위를 판단하기 어렵게 된다. 또한, 품질이 좋은 규칙을 만들고 유지보수하기 위해서는 전문가를 포함한 많은 인력이 필요하다.

이러한 규칙 기반 방법론의 단점을 보완하기 위해 최근 형태소 분석기를 포함한 자연어 처리 도구들은 빅데이터를 활용해 숨은 패턴을 찾아내는 통계 기반(statistical-based) 방법론을 사용한다. 통계 기반의 대표적인 한국어 형태소 분석기가 Mecab이며 대용량 한국어 학습 데이터를 기계 학습 모델을 이용해 사전 학습(Pre-train)시킨 분석기이다. 사전 학습이 완료된 Mecab 형태소 분석기는 새로운 텍스트에 대해 CRF(Conditional Random Field) 모델로 비지도(unsupervised) 학습을 하여 품사들의 연접(bigram)과 단어 비용을 학습하여 문장을 토큰화(tokenizing)하는 기준으로 활용하게 된다.

### 2.2.2 TF-IDF 가중치

TF-IDF(Term Frequency - Inverse Document Frequency) 가중치는 텍스트 마이닝에서 이용하는 가중치로, 어떤 단어가 특정 문서 내에서 얼마나 중요한 것인지를 나타내는 통계적 수치이며 문서의 핵심 키워드 추출, 문서

간 유사도 계산 등의 목적으로 사용될 수 있다. TF(단어 빈도, term frequency)는 특정한 단어가 문서 내에 얼마나 자주 등장하는지를 나타내는 값으로, TF의 값이 클수록 해당 단어가 문서에서 핵심적인 역할을 담당하는 것이라 볼 수 있다. 반면 단어 자체가 전체 문서군 내에서 자주 사용 되는 경우, 해당 단어가 흔하게 등장한다는 것을 의미한다. 이것을 DF(문서 빈도, document frequency)라고 하며, DF의 역수 형태를 IDF(역문서 빈도, inverse document frequency)라고 하며 다음과 같이 정의한다.

TF(d,t) : 특정 문서 d에서의 특정 단어 t의 등장 횟수

$$IDF(d, t) = \log\left(\frac{n}{1+df(t)}\right)$$

※ 여기서 n은 총 문서의 수, d는 특정 문서, t는 특정 단어, df(t)는 특정 단어 t가 등장한 문서의 수를 의미한다.

DF의 역수 형태인 IDF에 log가 사용된 이유는 총 문서의 수 n이 증가함에 따라 IDF의 값이 급격하게 커지는 것을 방지하기 위함이고, 분모에 1을 더하는 것은 특정 단어가 모든 문서에 한번도 등장하지 않는 경우(df(t)가 0인 경우)에 분모가 0이 되는 것을 방지하기 위함이다.

TF-IDF는 위에서 정의한 TF와 IDF를 곱한 값으로 정의된다. TF-IDF 가중치는 전체 문서에서 자주 등장하는 단어들에 대해서는 낮은 값을 가지게 되고, 특정 문서에서만 자주 등장하는 단어들에 대해서는 높은 값을 가지게

된다. 그래서 TF-IDF 가중치 계산을 통해 대부분의 문서에 등장하는 불용어의 중요도는 낮아지게 되고, 문서의 핵심이 되는 문장의 중요도는 높아지게 되는 효과를 기대할 수 있다.

### 2.2.3 코사인 유사도

TF-IDF를 통해 수치화한 문장 내 단어 벡터들의 코사인 유사도를 계산할 수 있다. 코사인 유사도란 벡터와 벡터 간의 유사도를 비교할 때 두 벡터 간의 사잇각을 구해서 얼마나 유사한지에 대해 수치로 나타낸 값이고 아래와 같이 정의한다.

[ n차원 벡터 A, B의 사잇각  $\theta$ 를 고려한 코사인 유사도 ]

$$\text{COS}(\theta) = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}}$$

코사인 유사도는 -1부터 1 사이의 범위를 가지고 서로 반대일 경우 -1, 서로 독립일 경우 0, 서로 유사할 경우 1에 가까운 값을 가지게 된다. 이와 같이 코사인 유사도를 통해 한 문서 내의 문장들의 유사성을 파악해 볼 수 있다.

### 2.2.4 설명 변수 생성

일전의 EDA 결과와 Mecab 형태소 분석기, TF-IDF 가중치, 코사인 유사도 등을 이용해 다음과 같은 설명 변수를 생성하였다.

1. Media\_name : 신문 기사를 작성한 신문사 이름
  2. Article\_len : 신문 기사의 길이
  3. Id : 신문 기사의 고유 ID로 모형 적합 후 각 신문 기사에서 상위 3개의 확률을 가지는 예측값을 핵심 문장으로 분류하는 과정에 이용
  4. First : 기사의 첫 번째 문장이면 1, 아니면 0 부여
  5. Percentile : 신문 기사 내 문장이 몇 % 지점인지 계산
  6. Bias : 각 신문사 별로 해당 위치를 전체 핵심 문장으로 선택할 확률
  7. Bias1 : 각 신문사 별로 해당 위치를 첫 번째 핵심 문장으로 선택할 확률
  8. Bias2 : 각 신문사 별로 해당 위치를 두 번째 핵심 문장으로 선택할 확률
  9. Bias3 : 각 신문사 별로 해당 위치를 세 번째 핵심 문장으로 선택할 확률
  10. SL : Mecab 형태소 분석기를 이용한 토큰화를 통해 각 문장에서 명사만 추출한 이후 
$$\frac{\text{한 문장 내 등장 단어 수}}{\text{신문 기사 내 가장 등장한 단어의 수가 많은 문장 단어 수}}$$
 계산
  11. TF-ISF : Document를 문장(Sentence)로 대체한 개념으로  
각 문장 내 단어들의 IF-IDF 가중치의 평균을 계산
  12. sent2sim : 신문 기사 내의 각 문장들의 단어 TF-IDF 벡터를 기반으로 문장 간의 코사인 유사도 계산
- 종속 변수는 Label으로 신문 기사 내에서 핵심 문장을 1, 아닌 문장을 0으로 인코딩 해주었다. 해당 과정을 거친 후의 설명 변수 데이터 프레임 형태

는 아래와 같다.

[ 설명 변수 생성 후 데이터 프레임 구조 ]

	media_name	article_len	id	first	bias	bias_1	bias_2	bias_3	SL	perc	TF_ISF	sent2sim
0	중부일보	11	7659	1	0.02	0.06	0.00	0.00	0.739130	0.0000	0.454083	0.184745
1	중부일보	11	7659	0	0.17	0.42	0.04	0.02	0.130435	0.1000	0.055564	0.158663
2	중부일보	11	7659	0	0.21	0.29	0.27	0.06	1.000000	0.2000	0.670845	0.205166
3	중부일보	11	7659	0	0.18	0.14	0.25	0.14	0.260870	0.3000	0.184990	0.221849
4	중부일보	11	7659	0	0.13	0.05	0.17	0.18	0.869565	0.4000	0.502019	0.208044
...	...	...	...	...	...	...	...	...	...	...	...	...
98203	머니투데이	7	6882	0	0.20	0.11	0.26	0.26	0.533333	0.1579	0.362183	0.139670
98204	머니투데이	7	6882	0	0.12	0.04	0.14	0.23	0.433333	0.2105	0.349388	0.138982
98205	머니투데이	7	6882	0	0.08	0.01	0.10	0.16	1.000000	0.2632	0.585797	0.124758
98206	머니투데이	7	6882	0	0.05	0.01	0.05	0.13	0.700000	0.3158	0.491999	0.128484
98207	머니투데이	7	6882	0	0.05	0.01	0.04	0.12	0.966667	0.3684	0.554197	0.125647

98208 rows x 12 columns

## 2.3 모형 구축

본 프로젝트에서 비교할 모형은 로지스틱 회귀 분석 모형(Logistic Regression), 나무 모형(Decision Tree), 랜덤 포레스트 모형(Random Forest), GBM(Gradient Boosting Model)을 발전시킨 XGBOOST 모형(eXtreme Gradient BOOSTing)이다. 각 모형들의 초모수(Hyperparameter)는 5-겹 교차 검증을 이용한 격자 탐색(Grid Search) 방식을 이용해 최적화 할 것이다. 모형 검정을 위해 훈련 데이터 셋과 테스트 데이터 셋을 8:2의 비율로 나누게 되는데 각 신문사의 패턴을 어느정도 모형에 학습시키기 위해 신문

사를 기준으로 층화 분할을 진행한다. 5-겹 교차 검증을 진행하기 때문에 훈련 데이터 셋을 별도로 나누어 줄 필요는 없으나, 이후에 적합할 XGBOOST 모형에서는 조기 종료(Early Stopping) 기능을 사용하기 위해 훈련 데이터 셋을 다시 8:2의 비율로 나누어 훈련 데이터의 20%를 검증 데이터(Validation Data)로 사용할 것이다. 필자는 분류 모형을 적합한 후 각 신문 기사의 문장 별로(id로 구분) 0부터 1 사이의 확률을 출력해 상위 3개의 확률을 가진 문장을 중요 문장(1)이라고 최종적으로 분류할 것이다. 분석에 사용하는 데이터는 중요하지 않은 문장이 80%에 해당하는 불균형 데이터인 점을 고려해 모형의 정밀도, 재현율을 기반으로 하는 Macro F1-score, Micro F1-score, ROC-AUC 세 가지 지표로 모형 평가 지표를 비교하도록 하겠다.

### 2.3.1 모형 구축을 위한 데이터 변환

모형에 범주형 수치가 들어가면 작동하지 않는 모형을 고려해 신문사 이름인 media\_name 변수를 원-핫 인코딩(One-Hot Encoding)해 더미(dummy) 변수를 생성한다. 또한 변수들의 척도가 다른 문제가 있어 표준화(standard scaling)를 진행해주었다.

### 2.3.2 순열 변수 중요도 ( Permutation Feature Importance )

모형 적합 전에 모형에 불필요한 변수를 선택하는 과정이 필요하다. 이러한 변수 선택의 과정에선 필자는 순열 변수 중요도를 고려한다. 순열 변수 중요도란 모델 예측에 가장 큰 영향을 미치는 Feature 를 파악하는 방법으로 적합된 모형을 통해 계산되고, 훈련된 모형이 특정 변수를 사용하지 않았을 때, 지표 성능 손실이 얼마나 발생하는지를 통해 변수의 중요도를 파악하는 방법이다. 여기서 특정 변수를 사용하지 않는다는 의미는 변수의 값을 무작위로 섞어서 노이즈(noise)로 만든다는 것을 의미한다.

순열 변수 중요도의 장점은 훈련된 모델과 데이터만 있으면 변수 중요도를 뽑을 수 있기 때문에 모델의 학습 과정, 내부 구조에 대한 정보가 없어도 적용할 수 있고 부분적 중요도(partial importance)가 아닌 다른 변수들과의 상호 작용이 포함된 중요도라는 점이다.

하지만 순열 변수 중요도를 사용함에 있어 주의해야 할 점 또한 있다. 데이터의 변수를 무작위로 섞기 때문에 실행마다의 결과가 상이할 수 있다는 점이다. 이와 같은 점을 보완하기 위해 반복 측정을 통해 일반화된 순열 변수 중요도를 측정할 필요가 있다. 본 프로젝트에서는 5번의 시뮬레이션을 통해 순열 변수 중요도 값을 측정했다.

### 2.3.3 로지스틱 회귀 분석

로지스틱 회귀분석은 종속 변수(Y)가 0 또는 1인 경우를 가정하고, 설명 변수의 값이 주어졌을 때의 Y의 분포가 베르누이 분포임을 가정하게 된다.

설명 변수(X)의 값이  $X = x$ 로 주어졌을 때의  $Y=y$ 의 확률은

$$P(Y = y \mid X = x) = p_x^y (1 - p_x)^{1-y} \text{ where } y = 0 \text{ or } 1$$

로 정의되고 본 프로젝트에서의  $p_x$ 는 중요 문장일 확률(1)이 된다.

만약 선형 확률 모형  $p_x = \alpha + \beta x$ 를 고려하면 확률의 값이 0보다 작거나 1보다 큰 경우가 발생할 수 있고, 모수  $\alpha, \beta$ 의 LSE가 선형 불편 추정량에 있어서 최소 분산을 갖지 않게 된다. 그래서 로지스틱 회귀 분석에서는 비선형 확률 모형인  $p_x = \frac{\exp(\alpha + \beta x)}{1 + \exp(\alpha + \beta x)}$ 를 가정하고, 최종적으로  $\text{logit}(p_x) = \ln\left(\frac{p_x}{1-p_x}\right) = \alpha + \beta x$ 로 나타내어지는 로짓 모형(logit model)을 정의한다. 로짓 모형에서의 가능도함수는 데이터 개수만큼의 i.i.d한 베르누이 분포의 곱으로 나타내어지고, Newton-Raphson 방법을 통해 로지스틱 회귀 분석의 해를 최적화하게 된다.

모형을 적합하기 이전에 기본 모형(default setting model)을 적합한 후 순열 변수 중요도를 살펴본다. 순열 변수 중요도가 Weight라는 값으로 나타나게 되는데, 만약 추정치가 음수이고 신뢰구간이 0을 포함하지 않으면 모형 적합에 방해가 되는 변수라고 판단해 제거해 주었다.



[ 로지스틱 회귀 분석 모형 순열 변수 중요도 ]

Weight	Feature
0.2459 ± 0.0077	bias_1
0.1608 ± 0.0046	bias_2
0.0843 ± 0.0070	bias
0.0157 ± 0.0022	article_len
0.0139 ± 0.0031	SL
0.0131 ± 0.0030	media_name_매일경제
0.0095 ± 0.0024	TF_ISF
0.0073 ± 0.0040	media_name_디지털타임스
0.0072 ± 0.0024	media_name_아주경제
0.0064 ± 0.0009	bias_3
0.0037 ± 0.0022	sent2sim
0.0027 ± 0.0018	perc
0.0023 ± 0.0016	media_name_머니투데이
0 ± 0.0000	first
-0.0000 ± 0.0003	id
-0.0017 ± 0.0031	media_name_중부일보

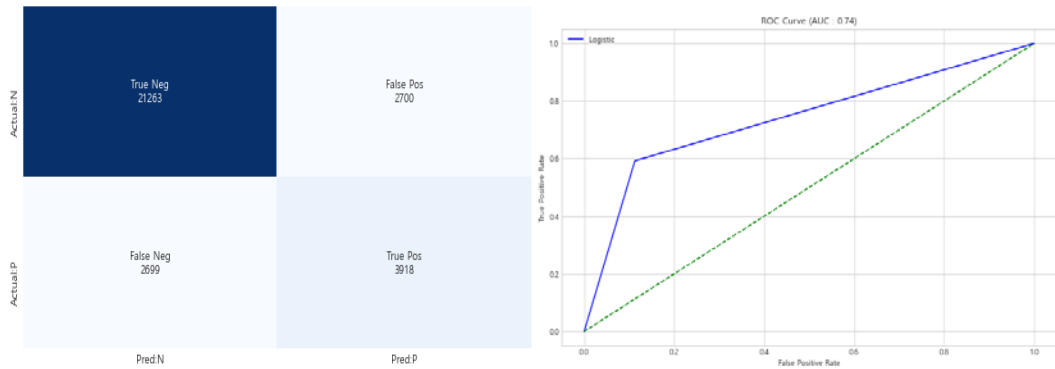
로지스틱 회귀 분석 모형에서는 순열 변수 중요도가 음수이고 신뢰구간이 0을 포함하지 않는 변수가 없었기에 변수를 제거하지 않았다.

필자는 능형 회귀(ridge regression)에 주로 사용되는 L2-벌점을 로지스틱 회귀 분석에 적용하여 모형의 과적합을 방지하고자 하였고, 격자 탐색을 통해 L2 벌점의 가중치를 나타내는 최적의 초모수  $C = \frac{1}{\lambda}$  는 0.1로 결정했다. 테스트 셋으로 지표들을 평가한 결과 F1 score(Micro)는 0.823, F1 score(Macro)는 0.74, ROC-AUC는 0.739로 나타났다. 중요 문장(1)에 대해 잘 분류하는지에 대한 지표로는 정밀도, 재현율을 고려할 수 있다. 혼동 행렬을 통해 대략적으로 절반이 조금 넘게 중요 문장(1)을 잘 분류해내는 것을 알 수 있다. 결과는 아래와 같다.

## [ 로지스틱 회귀 분석 모형 평가 지표 ]

```
Test:
Precision score / Recall Score / F1 score(micro) / F1 score(macro) / PR-AUC / ROC-AUC
0.59202          / 0.59211          / 0.82345          / 0.73971          / 0.6362          / 0.73972
Test confusion matrix
TEST ACCURACY : 0.82
```

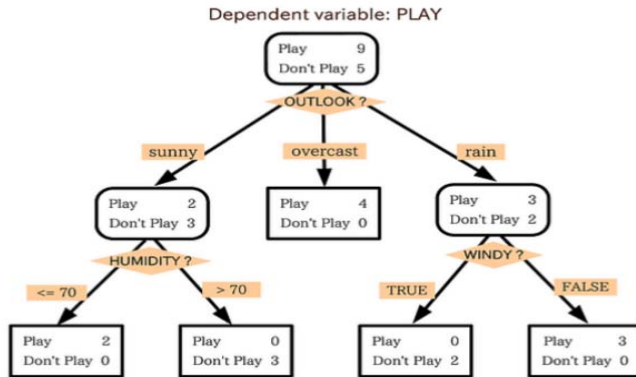
## [ 로지스틱 회귀 분석 모형 혼동 행렬 및 ROC curve ]



### 2.3.4 의사 결정 나무

의사 결정 나무 모형은 설명 변수들을 사용하여 예측 가능한 규칙들의 집합을 생성하는 알고리즘이다. 의사결정나무는 분류(classification)와 회귀(regression) 두 경우에 모두 적용이 가능하고, 끝 마디(terminal node)에서 가장 빈도가 높은 범주로 새로운 데이터를 분류하거나, 끝 마디에 속하는 데이터들의 값 평균으로 데이터를 예측하게 된다. 분류 나무 모형의 나무 그림 예시는 다음과 같다.

[분류 나무 모형의 나무 그림 예시 그림 ]



해당 예시는 운동 경기가 열리면 Play, 열리지 않으면 Don't Play로 분류하는 이진 분류 문제의 나무 모형 그림이다. 해당 예시에서는 날씨가 맑고 습도가 70 아래이면 Play인 예시가 2개, Don't Play인 예시가 0개 있어 해당 조건을 만족하는 데이터를 Play로 분류하게 될 것이다.

데이터를 분할하는 변수 조건을 나누는 기준은 불순도(impurity)를 계산하는 엔트로피(Entropy)와 지니 계수(Gini Index)로 나눌 수 있다. 각각의 식은 다음과 같다.

[ A 영역의 엔트로피 ]

$$\text{Entropy}(A) = -\sum_{k=1}^m p_k \log_2 p_k :$$

where  $p_k$  : A 영역 레코드 중 k 범주 안의 레코드 비율

[ A 영역의 지니 계수 ]

$$G.I(A) = 1 - \sum_{k=1}^m p_k^2$$

엔트로피와 지니 계수가 작을수록 불순도가 감소해 정보를 획득하게 되는 것이라고 볼 수 있다. 필자는 격자 탐색을 통해 나무 모형의 불순도 측도인 엔트로피와 지니 계수 중 더 좋은 불순도 측도를 찾고자 하였으며, 중단 노드의 최소 샘플 수, 나무의 깊이에 관련한 초모수를 조정해주었다.

모형을 적합하기 이전에 기본 모형(default setting model)을 적합한 후 순열 변수 중요도를 살펴본 결과는 아래와 같다.

[ 의사 결정 나무 모형 순열 변수 중요도 ]

Weight	Feature
0.1682 ± 0.0047	bias
0.0399 ± 0.0086	TF_ISF
0.0379 ± 0.0032	article_len
0.0325 ± 0.0088	SL
0.0240 ± 0.0020	bias_1
0.0225 ± 0.0029	bias_3
0.0191 ± 0.0047	sent2sim
0.0108 ± 0.0018	bias_2
0.0023 ± 0.0009	media_name_디지털타임스
0.0022 ± 0.0022	perc
0.0018 ± 0.0058	id
0.0016 ± 0.0005	media_name_중부일보
0.0005 ± 0.0010	media_name_머니투데이
0.0005 ± 0.0002	first
0.0001 ± 0.0002	media_name_매일경제
0.0000 ± 0.0009	media_name_아주경제

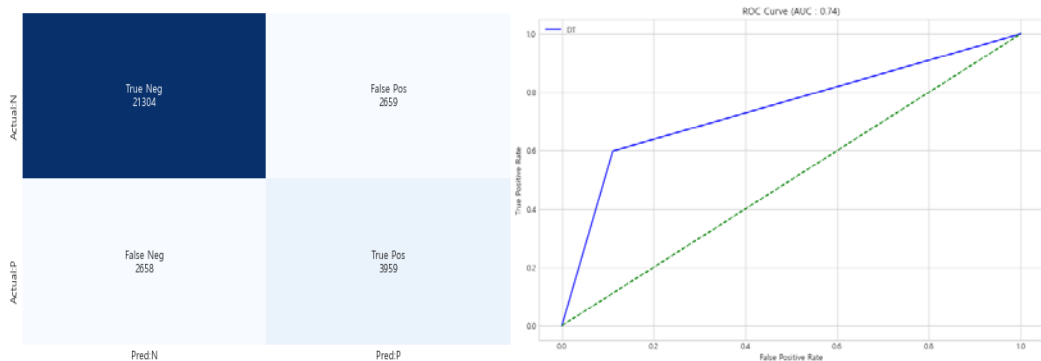
의사 결정 나무 모형에서는 순열 변수 중요도가 음수이고 신뢰구간이 0을 포함하지 않는 변수가 없었기에 변수를 제거하지 않았다.

격자 탐색의 결과 엔트로피 불순도, 중단 노드의 최소 샘플 수는 3, 나무의 깊이는 6으로 초모수를 결정했다. 테스트 셋으로 지표들을 평가한 결과 F1 score(Micro)는 0.826, F1 score(Macro)는 0.744, ROC-AUC는 0.744로 나타났다. 그 결과는 아래와 같다.

## [ 의사 결정 나무 모형 평가 지표 ]

```
Test:
Precision score / Recall Score / F1 score(micro) / F1 score(macro) / PR-AUC / ROC-AUC
0.59822          / 0.59831          / 0.82613          / 0.74366          / 0.64172          / 0.74367
Test confusion matrix
TEST ACCURACY : 0.83
```

## [ 의사 결정 나무 모형 혼동 행렬 및 ROC curve ]



### 2.3.5 랜덤 포레스트

배깅(bagging) 방법론에서는 모든 변수를 사용하여 의사 결정 나무를 기반으로 하는 약한 학습기를 여러 개 만들어서 결과를 앙상블하게 된다. 앙상블의 방법은 크게 두 가지로 나뉘게 되는데 직접 투표(Hard Voting)의 경우, 학습기마다 예측한 범주들에 대해 빈도가 가장 높은 범주로 최종 할당하는 방식이다. 간접 투표(Soft Voting)의 경우, 학습기마다 예측 데이터에 대해 각 범주에 속할 확률을 예측한 값의 평균 내어, 이 평균 확률이 가장 큰 범주로

최종 할당하는 방식이다. 랜덤 포레스트의 경우, 직접 투표의 한 종류인 다수결 투표(Majority voting) 방식을 사용하여 다수결로 결과를 취합하게 된다.

이 때, 모든 변수를 사용해 학습기를 훈련시키게 되면 영향력이 큰 특정 변수를 중심으로 나무 모형이 분기될 가능성이 높고, 학습기 간의 상관관계가 높아지게 된다는 문제점이 발생한다. 학습기 간의 상관관계가 높다는 배경의 단점을 보완하기 위해 랜덤 포레스트 모형에서는 전체 변수 중 샘플링(sampling)된 변수들로 나무 학습기들을 생성한다. 그래서 다양한 변수의 특성을 학습하는 학습기들을 만들 수 있어 학습기 간의 상관관계를 배경에 비해 줄일 수 있게 된다. 필자는 랜덤 포레스트 모형에서 격자 탐색을 통해 엔트로피와 지니 계수 중 더 좋은 불순도 측도를 찾고자 하였으며, 나무 기반 학습기의 학습에 사용하는 변수의 개수, 나무 기반 학습기의 깊이에 관련한 초모수를 조정해주었다.

모형을 적합하기 이전에 기본 모형(default setting model)을 적합한 후 순열 변수 중요도를 살펴본 결과는 아래와 같다.

[ 랜덤 포레스트 모형 순열 변수 중요도 ]

Weight	Feature
0.2189 ± 0.0092	bias
0.1246 ± 0.0033	bias_1
0.0357 ± 0.0030	TF_ISF
0.0355 ± 0.0058	bias_2
0.0117 ± 0.0036	sent2sim
0.0088 ± 0.0063	SL
0.0057 ± 0.0032	article_ten
0.0025 ± 0.0044	perc
0.0006 ± 0.0008	media_name_중부일보
0.0003 ± 0.0015	media_name_아주경제
0.0000 ± 0.0017	media_name_매일경제
-0.0009 ± 0.0014	media_name_디지털타임스
-0.0022 ± 0.0060	id
-0.0024 ± 0.0024	media_name_머니투데이
-0.0028 ± 0.0012	first
-0.0091 ± 0.0037	bias_3

랜덤 포레스트 모형에서는 bias\_3, first, media\_name\_머니투데이 변수의 순열 변수 중요도 추정치가 음수이고 신뢰구간이 0을 포함하지 않아 모형 적합 전에 제거해 주었다.

격자 탐색의 결과 엔트로피 불순도, 나무 기반 학습기의 학습에 사용하는 변수의 개수는 전체 변수 개수의 1/2배, 나무의 깊이는 6으로 초모수를 결정했다. 테스트 셋으로 지표들을 평가한 결과 F1 score(Micro)는 0.828, F1 score(Macro)는 0.747, ROC-AUC는 0.747로 나타났다. 그 결과는 다음과 같다.

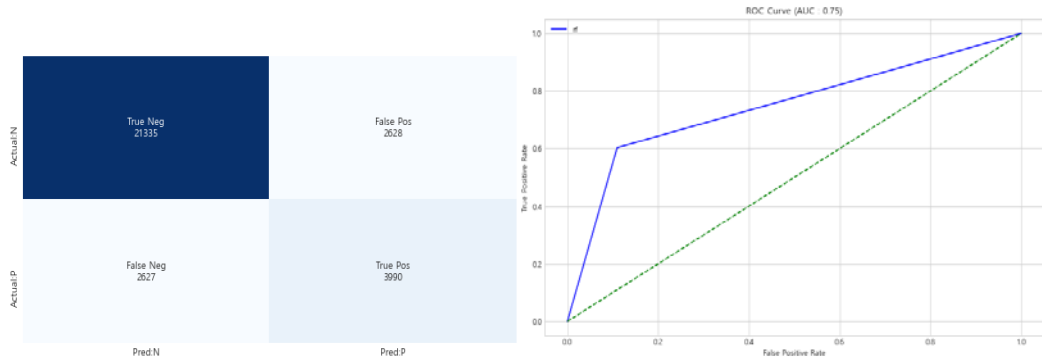
[ 랜덤 포레스트 모형 평가 지표 ]

```

Test:
Precision score / Recall Score / F1 score(micro) / F1 score(macro) / PR-AUC / ROC-AUC
0.6029          / 0.60299          / 0.82816          / 0.74665          / 0.6459          / 0.74666
Test confusion matrix
TEST ACCURACY : 0.83

```

## [ 랜덤 포레스트 모형 혼동 행렬 및 ROC curve ]



### 2.3.6 XGBOOST ( eXtreme Gradient BOOSTing )

부스팅 기반 모형인 GBM(Gradient Boosting model)은 병렬적으로 학습을 하는 배깅 방식과 달리 순차적으로 모델을 학습한다. 그래서 모형 학습 속도가 느린 편이고, 부스팅 모형은 잔차를 줄여나가는 방향으로 학습을 해 편향이 작고 분산이 큰(Low Bias, High Variance) 모형을 만들어져 과적합이 될 가능성이 높다.

그래서 새롭게 등장하는 XGBOOST, LightGBM과 같은 모형들은 이러한 단점을 해결하기 위한 기법들을 알고리즘에 포함시키게 된다. XGBOOST의 경우 병렬 병렬 학습을 지원하여 GBM의 느린 수행 속도를 보완한다. 또한 과적합을 방지하고자 L1, L2 규제를 손실함수에 적용하고 나무 기반 학습기들에 대한 가지치기(pruning)을 모형 적합 시 알고리즘 내에서 자동적으로 수



행한다. 그에 더해 일정 기간 동안 모형 지표의 유의미한 발전이 없으면 학습을 중단시키는 조기 중단(Early Stopping) 기능을 설정할 수 있다.

필자는 XGBOOST 모형에서 격자 탐색을 통해 중단 노드의 최소 샘플 수, 나무 기반 학습기의 깊이에 관련한 초모수를 조정하고자 했고, 500번의 가중치 갱신 과정에서 50번 동안 모델 지표의 유의미한 발전이 없으면 학습을 중단하는 조기 중단을 설정해 모형 학습을 진행했다.

모형을 적합하기 이전에 기본 모형(default setting model)을 적합한 후 순열 변수 중요도를 살펴본 결과는 아래와 같다.

[ XGBOOST 모형 순열 변수 중요도 ]

Weight	Feature
0.3457 ± 0.0097	bias
0.0496 ± 0.0022	bias_1
0.0410 ± 0.0063	TF_ISF
0.0211 ± 0.0060	sent2sim
0.0184 ± 0.0040	article_len
0.0165 ± 0.0020	bias_2
0.0115 ± 0.0071	SL
0.0015 ± 0.0026	bias_3
0.0012 ± 0.0013	id
0.0009 ± 0.0006	first
0.0007 ± 0.0004	media_name_중부일보
0.0003 ± 0.0005	media_name_디지털타임스
-0.0000 ± 0.0003	media_name_매일경제
-0.0004 ± 0.0007	media_name_머니투데이
-0.0011 ± 0.0008	media_name_아주경제
-0.0029 ± 0.0019	perc

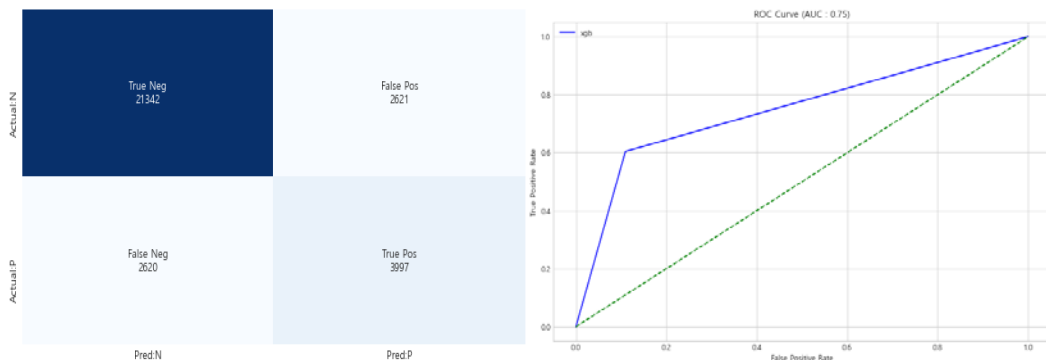
XGBOOST 모형에서는 perc, media\_name\_아주경제 변수의 순열 변수 중요도 추정치가 음수이고 신뢰구간이 0을 포함하지 않아 모형 적합 전에 제거해주었다.

격자 탐색의 결과 종단 노드의 최소 샘플 수는 6, 나무의 깊이는 3으로  
초모수를 결정했다. 테스트 셋으로 지표들을 평가한 결과 F1 score(Micro)는  
0.829, F1 score(Macro)는 0.747, ROC-AUC는 0.747로 나타났다. 그 결과는  
아래와 같다.

[ XGBOOST 모형 평가 지표 ]

```
Test:
Precision score / Recall Score / F1 score(micro) / F1 score(macro) / PR-AUC / ROC-AUC
0.60396          / 0.60405          / 0.82861          / 0.74732          / 0.64684          / 0.74734
Test confusion matrix
TEST ACCURACY : 0.83
```

[ XGBOOST 모형 혼동 행렬 및 ROC curve ]



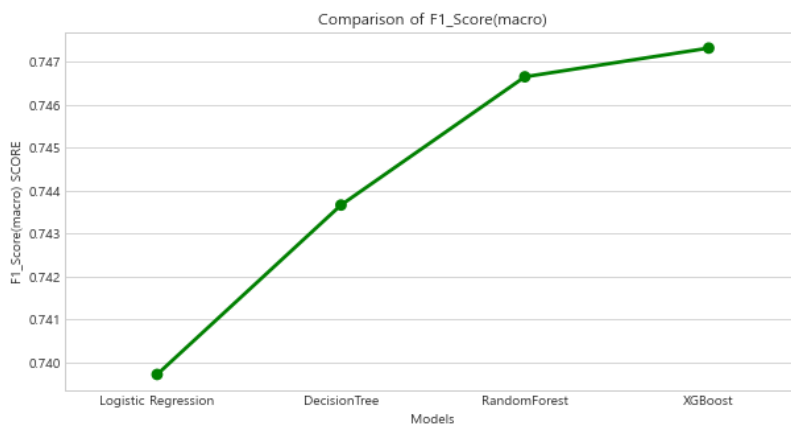
## 제 3 장. 결론

---

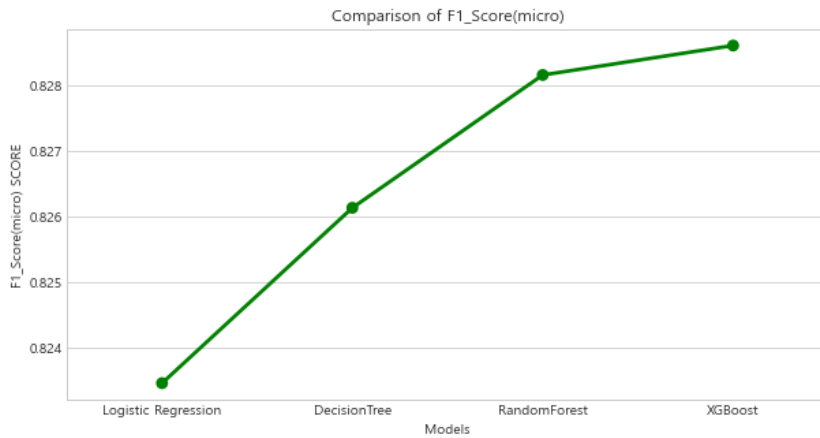
### 3.1 분석 결과

앞서 언급했듯이 본 프로젝트의 데이터가 중요하지 않은 문장이 80%에 해당하는 불균형 데이터인 점을 고려해 모형의 정밀도, 재현율을 기반으로 하는 Macro F1-score, Micro F1-score, ROC-AUC 세 가지 지표로 모형 평가 지표를 비교하도록 하겠다. 로지스틱 회귀 분석 모형, 나무 모형, 랜덤 포레스트 모형, XGBOOST 모형을 비교한 결과 모든 모형 평가 지표에서 XGBOOST 모형이 가장 좋았고, 로지스틱 회귀 분석이 결과가 좋지 않았다. 자세한 사항은 아래 첨부된 그림들을 통해 살펴볼 수 있다.

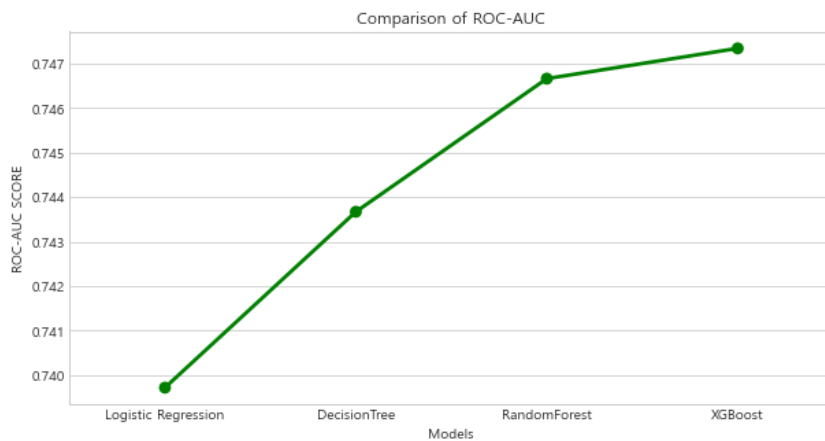
[ 사용 모형들의 F1\_score(Macro) 평가 지표 비교 ]



[ 사용 모형들의 F1\_score(Micro) 평가 지표 비교 ]



[ 사용 모형들의 ROC-AUC 평가 지표 비교 ]



위의 결과를 보았을 때, 신문 기사 내 중요한 문장을 추출하는 이진 분류 모형을 구축할 때 네 개의 모형 중에 XGBOOST가 가장 문장이 중요한 경우(1)도 잘 분류하고 문장이 중요하지 않은 경우(0)도 잘 분류한다는 결론을 낸다.

### 3.2 추후 연구 방향

본 프로젝트에서는 AI HUB의 추출 요약 데이터를 이용하여 추출 요약 문제를 신문 기사 내 중요한 문장인지 아닌지를 분류하는 이진 분류 문제로 바꾸어 풀어보았다. 필자는 데이터의 불균형에 초점을 맞춰 F1 점수와 AUC-ROC로 모델을 평가하였으나, 추출 요약을 평가하는 일반적인 평가지표는 ROUGE-1,2,3으로 모델이 예측한 추출요약문과 사람이 생성한 요약문의 unigram, bigram, trigram이 겹치는 정도를 나타낸다. 추후에는 해당 지표를 최적화하는 방향으로 연구를 진행할 수도 있을 것이다. 또한 추출 요약에 사용되는 전통적인 통계 모형인 텍스트 랭크(TextRank) 모형이나 BERT 계열의 자연어 추출 요약 모형을 사용해 볼 수도 있을 것이다. 특히 BERT 계열의 모형은 대용량 자연어 데이터 셋을 훈련시켜 자연어 생성, 요약 등에 최적화된 모형으로 해당 모형의 추출 요약 결과와 기계 학습 모형의 추출 요약 결과를 비교해보는 연구도 흥미로울 것이다.

## 제 4 장. 부록

---

### 4.1 참고 문헌

Celso Kaestner, Automatic Text Summarization Using a Machine Learning Approach, Brazilian Symposium on Artificial Intelligence

Andrew P. Bradley, The use of the area under the ROC curve in the evaluation of machine learning algorithms, ScienceDirect

스카이 요시노리, 한국어 자동 형태소 분석 사전의 개발, 한국사전학회  
이종화, 이문봉, 김종원, TF-IDF를 활용한 한글 자연어 처리 연구, 한국정보시스템학회

Pinky Sitikhu, Kritish Pahi, Pujan Thapa, Subarna Shakya, A Comparison of Semantic Similarity Methods for Maximum Human Interpretability, arXiv

André Altmann, Laura Toloşi, Oliver Sander, Thomas Lengauer, Permutation importance: a corrected feature importance measure, Bioinformatics

Leo Breiman, Random Forests, SpringerLink

Tianqi Chen, Carlos Guestrin, XGBoost: A Scalable Tree Boosting System, arXiv