# 4. LLM 선호도 예측 분류 모델 개발 (Kaggle Competition 기반)

## 프로젝트 개요

사용자의 응답 데이터를 기반으로 두 가지 LLM 모델(Model A vs Model B) 중 어떤 응답을 선호할지 예측하는 **다중 분류(Multi-Class Classification)** 모델을 개발했습니다. (노트북: LLM\_선호도\_예측\_업그레이드\_모델.ipynb, new-session-for-llm-from-salmonyeonwoo.ipynb 통합)

## ⚙️ 사용 기술 및 역할

* **주요 기술:** Python, Pandas, Scikit-learn (MinMaxScaler, MultinomialNB, SVC, RandomForestClassifier), TF-IDF
* **핵심 역할:**
  1. **고급 특징 엔지니어링:** 텍스트 응답에 **TF-IDF 벡터화**를 적용하여 텍스트 특징을 추출하고, 수치형 특징(토큰 수, 구두점 수)과 결합하여 예측력을 높임.
  2. **데이터 전처리:** Multinomial Naive Bayes 적용을 위해 특징 데이터에 **MinMaxScaler**를 적용하여 $[0, 1]$ 범위로 스케일링하는 문제 해결 로직 구현.
  3. **모델 비교 분석:** MNB, SVC, RandomForestClassifier를 적용하고 Log Loss 지표로 성능을 비교 분석.

## 💡 핵심 분석 및 결과

1. **Feature Engineering 성과:** 텍스트 특징(TF-IDF)과 수치형 특징을 성공적으로 결합하여, 모델이 텍스트 내용과 응답 길이/구조를 모두 고려하도록 특징 공간을 확장.
2. **모델 학습 및 스케일링 문제 해결:**
   * MNB 모델은 입력 특징이 음수여서 초기 학습에 실패했으나, **MinMaxScaler를 통해 데이터를 전처리**하여 이 문제를 해결하고 학습에 성공.
   * **Multinomial Naive Bayes (MNB):** 훈련 데이터에서 Log Loss **0.0119**를 달성 (높은 훈련 성능 확인).
3. **도전 과제:** 데이터셋의 **심각한 클래스 불균형** 및 작은 데이터 크기로 인해 안정적인 교차 검증 및 하이퍼파라미터 튜닝의 어려움을 보고하고, 추가 데이터 확보의 필요성을 제시.

## 📌 주요 코드 로직 (TF-IDF 및 스케일링 추상화)

from sklearn.feature\_extraction.text import TfidfVectorizer  
from sklearn.preprocessing import MinMaxScaler  
from sklearn.naive\_bayes import MultinomialNB  
  
# 1. 텍스트 특징 추출 및 결합  
texts = df['response\_a'].fillna('') + ' ' + df['response\_b'].fillna('')  
tfidf\_vectorizer = TfidfVectorizer(max\_features=1000)  
tfidf\_features = tfidf\_vectorizer.fit\_transform(texts)  
# numerical\_features와 결합하여 최종 특징 X 생성  
  
# 2. MNB를 위한 데이터 스케일링 (음수값 방지)  
scaler = MinMaxScaler()  
X\_scaled = scaler.fit\_transform(X) # X는 최종 특징 행렬  
  
# 3. 모델 학습 및 예측  
model = MultinomialNB()  
model.fit(X\_scaled, y)   
# log\_loss(y\_test, model.predict\_proba(X\_test\_scaled)) 계산 및 결과 검증