

# Progress Report

2020170837 최원준 | 2021320328 장지웅 | 2022320157 노규주

## 1. Problem Definition

현대 금융시장은 주요 정치인, 기업가 등 영향력 있는 인물들의 발언에 즉각적이고 민감하게 반응한다. 제롬 파월 연준 의장의 연설이나 일론 머스크의 트윗 하나가 금리, 환율, 주가 등 주요 경제 지표의 단기 변동성을 촉발하는 사례가 빈번하게 관찰된다. 이러한 발언들은 기존 거시경제 지표만으로는 설명하기 어려운 새로운 정보 흐름을 형성하며, 급등락이 특징인 금융 시장의 단기 움직임을 가늠할 수 있는 중요한 신호가 된다. 따라서 이러한 텍스트 데이터를 정량적으로 분석하여 금융 지표 변동과의 연관성을 규명하고 예측 모델을 구축하는 것은, 시장의 비선형적 움직임을 이해하고 리스크를 관리하는 데 중요한 학술적, 실용적 가치를 지닌다.

본 프로젝트의 최종 목표는 **텍스트 데이터와 국제 경제 지표 변동 간의 연관성을 규명하고, 이를 예측하는 머신러닝 모델을 구축하는** 것이다. 기존 proposal 단계에서는 이러한 배경 하에 ‘영향력 있는 인물의 SNS 발언’을 핵심 분석 대상으로 설정했다. SNS는 정보 전달 속도가 매우 빨라, 시장의 단기 변동성을 포착할 수 있는 중요한 신호로 판단했기 때문이다.

그러나 프로젝트 초기 데이터 수집 단계에서 두 가지 현실적인 제약이 있었다. 첫째, 주요 SNS 플랫폼의 크롤링 방지 정책으로 인해, 실험에 필요한 과거 데이터를 안정적으로 대량 수집하는 것이 기술적으로 불가능함을 확인했다. 둘째, 분석 대상으로 설정했던 ‘2018년 TIME지 선정 영향력 있는 100인’의 실제 SNS 활용 비율이 인물별로 매우 큰 편차를 보였다. 일부 인물은 SNS를 거의 사용하지 않아, 모든 인물을 동일 선상에서 비교 분석하고 일관성 있는 영향력 모델을 구축하는데 한계가 있었다.

이러한 데이터 수집 및 일관성 문제를 해결하기 위해, 분석 대상을 개인의 SNS 발언에서 ‘**주요 언론사의 해당 인물 관련 뉴스 기사**’로 변경하는 것이 목표 달성을 더 적합하다고 판단했다. 뉴스 기사는 SNS 데이터보다 수집이 용이할 뿐만 아니라, 1차적으로 정제된 정보를 담고 있어 시장 참여자들의 반응을 보다 안정적으로 반영할 수 있다는 장점이 있다.

수정된 문제 정의에 따라, 뉴스 기사 텍스트에 대한 feature engineering을 본 프로젝트의 핵심 기술 과제로 설정했다. 단순히 기사 텍스트를 입력하는 것을 넘어, 기사가 내포한 의미와 시장 파급력을 정량화된 feature로 추출하는 데 집중할 계획이다. 본 progress report에서는 이 목표를 달성하기 위해 시도한 다양한 방법론과 그 중간 결과를 제시한다.

1. **텍스트 임베딩:** 먼저, 뉴스 기사의 headline과 body text가 가진 의미론적 정보를 벡터로 변환하기 위해 gte 및 miniLM과 같은 사전 학습된 경량 언어 모델을 활용한다. 텍스트를 처리하는 방식(e.g., chunking)에 따른 성능 변화도 함께 실험한다.
2. **차원 축소:** 텍스트 임베딩으로 생성된 고차원 feature 벡터를 그대로 사용하는 경우와, PCA와 같은 기법을 통해 차원 축소를 적용한 경우의 모델 성능을 비교하여, 텍스트 정보의 효율적인 압축 방안을 탐색한다.
3. **머신러닝 모델 적용:** 추출된 임베딩 벡터와 금융 지표를 결합하여 S&P500 지수를 예측한다.
  - a. **baseline:** 텍스트 데이터의 유효성을 검증하기 위해, 금융 지표만으로 미래를 예측하는 **Linear regression** 과 전통적인 시계열 모델인 **SARIMA**를 baseline으로 설정한다.
  - b. **core:** features 사이 복잡한 비선형 관계 및 시계열 특성을 학습하기 위해, 양상블 기법인 **LightGBM**과 순환 신경망(RNN) 기반의 딥러닝 모델인 **GRU**를 적용하여 성능을 비교 분석한다.

이러한 단계적 접근을 통해, 뉴스 기사 텍스트가 실제 금융 시장 예측에 유의미한 시그널을 제공할 수 있는지 검증하고, 가장 효과적인 feature engineering 및 모델 조합을 탐색하고자 한다.

## 2. Methodology and Experimental Approach

### 2.1 Data Selection and Collection

본 프로젝트에서는 국제 경제 지표 변동과 뉴스 텍스트 간의 상관성을 분석하기 위해, 글로벌 금융시장을 대표하는 S&P500 지수를 주요 예측 대상으로 설정하였다. S&P500은 미국 상장기업의 시가총액을 기반으로 산출되는 종합 주가지수로, 전 세계 경기 흐름과 투자심리를 가장 잘 반영하는 지표 중 하나이다.

분석 기간은 2017년부터 2019년까지로 한정하였다. 이는 코로나19 발생 이전 시기로, 팬데믹으로 인한 비정상적 급등락을 배제하면서도 트럼프 행정부의 정책 발언, 금리 조정, 무역 분쟁 등으로 인해 시장 변동성이 활발히 나타났던 구간이다. 따라서 뉴스 텍스트와 시장 지표 간의 단기적 상관관계를 비교적 안정적으로 관찰할 수 있다. 텍스트 데이터는 The Guardian의 기사 데이터를 기반으로 수집하였다. 분석 대상 인물은 TIME지의 100 Most Influential People of 2018 리스트[\[1\]](#)를 기준으로 선정하였다. 해당 인물들은 2017-2019년 기간 동안 국제 사회와 금융시장 전반에 걸쳐 높은 영향력을 지닌 주요 정치인·기업인·문화예술인으로 구성되어 있다.

데이터 수집은 The Guardian의 오픈 API[\[2\]](#)를 활용하여 진행하였다.

- 각 인물 이름을 키워드로 설정하여 2017-2019년 사이 게시된 모든 관련 기사를 크롤링하였다.
- 수집된 데이터에는 발행일(*webPublicationDate*), 제목(*headline*), 본문(*bodyText*), 단어 수(*wordcount*), 태그(*tags\_titles*, *tags\_types*) 등이 포함된다.

이 과정을 통해 총 약 23만 건의 기사 데이터를 확보하였다. 연도별 기사 수는 2017년이 가장 많았으며, 인물별로는 Donald Trump, Jeff Sessions, Ryan Coogler 등의 기사량이 두드러졌다. 수집된 데이터의 연도별·인물별 분포에 대한 상세 통계는 [Table 1]과 [Table 2]에서 확인할 수 있다. 본 데이터셋은 인물별·시기별 뉴스 텍스트를 통해 사회적 이슈와 시장 반응을 연결할 수 있는 기반 자료로 활용되며, 이후 단계에서 텍스트 임베딩 및 시계열 예측 모델링의 입력 데이터로 사용된다.

Statistics	2017	2018	2019	Total
Average	807.89	756.18	761.76	2325.83
Sum	80789	75618	76176	232583

[Table 1] 연도별 수집 뉴스 기사 통계

person	2017	2018	2019	total
Donald Trump	10669	8174	7700	26543
Jeff Sessions	3224	2870	2594	8688
Ryan Coogler	2731	2502	2209	7442

[Table 2] 연도별 기사량이 가장 많은 주요 인물 통계

### 2.2 Text Embedding and Data Representation

기사들의 의미를 벡터로 표현하기 위해 text embedding 기법을 적용하였다. 본 프로젝트의 목적은 단어의 단순 빈도가 아닌, 뉴스 문장에서 드러나는 의미적 유사성과 문맥적 관계를 반영하는 것이다. 따라서 Bag-of-Words나 TF-IDF와 같은 전통적인 통계 기반 표현 방식은 사용하지 않았다. 뉴스 기사는 일반적으로 본문이 길고 구조가 복잡하며, 주제 전환이나 인용문 등 다양한 문맥적 변화를 포함한다. 이러한 특성을 반영하기 위해 여러 임베딩 단위를 실험적으로 적용하였고, 최종적으로 총 6가지 임베딩 방식을 설계하였다. headline과 bodytext 각각을 대상으로 하여 다양한 입력 단위에서 의미 정보를 포착할 수 있도록 하였다.

사용한 텍스트 임베딩 모델은 두 가지이다.

- all-MiniLM-L12-v2[\[3\]](#)

- Sentence Transformers 계열의 경량 모델로, 문장 수준의 의미 유사도 계산에 효율적이다. Transformer 기반 구조를 사용하며, SBERT(Sentence-BERT)의 개량형으로 문장 임베딩을 빠르게 생성하면서도 의미적 일관성을 유지한다. 해당 모델의 출력 벡터 차원은 384이며, 상대적으로 적은 파라미터 수로 인해 대규모 뉴스 데이터 처리에 유리하다. 본 프로젝트에서는 headline과 bodytext 임베딩 모두에 활용되었다.
- **Alibaba-NLP/gte-large-en-v1.5[4]**
  - 대규모 문맥 정보를 학습한 고성능 언어모델로, 장문 텍스트의 의미적 구조를 세밀하게 반영할 수 있다. Transformer 기반으로 대량의 웹 및 뉴스 데이터를 학습하였으며, semantic textual similarity task에서 높은 성능을 보인다. 해당 모델의 출력 벡터 차원은 1024로, 본문 내에서 등장하는 여러 인물, 사건, 정책 간의 의미적 연결성을 더 풍부하게 표현할 수 있다.

임베딩 방식은 다음과 같이 구성하였다.

### 1. Headline embedding

- 각 기사의 headline을 단일 문장으로 임베딩하여, 기사 주제의 핵심 의미를 벡터로 표현하였다.

### 2. Bodytext embedding

- 기사 본문이 모델의 최대 입력 토큰 길이를 초과하는 경우, 토큰 제한까지 입력한 후 나머지 텍스트를 추가로 분할하여 각각 임베딩하였다. 이후 분할된 모든 벡터를 mean pooling하여 기사 전체를 대표하는 단일 벡터로 통합하였다. 이 방식은 모델의 입력 한계를 고려하면서도, 긴 기사에서 주요 의미가 손실되지 않도록 설계되었다.

### 3. Bodytext chunking + overlap pooling embedding

- 본문을 일정 길이 단위로 나누되, 각 구간 간 overlap을 포함하여 chunk 단위로 임베딩하였다. overlap을 적용함으로써 문단 경계에서 발생할 수 있는 문맥 단절을 최소화하였으며, 생성된 chunk-level 벡터는 mean pooling을 통해 기사 단위 벡터로 통합하였다. 이 방식은 문맥 연속성을 유지하면서 긴 텍스트의 전체 의미를 안정적으로 반영한다.
- gte-large-en-v1.5를 적용하는 방식은 코드 실행 중 오류로 인해 완전한 임베딩이 아직 완료되지 않았으며, 이는 추후 구현 과제로 남겨두었다.

이와 같은 다양한 임베딩 전략을 통해 뉴스 기사의 문맥적 의미, 주제적 흐름, 그리고 감정적 뉴앙스를 동시에 반영하는 벡터 표현을 구축하였다. 이러한 벡터는 이후 단계의 Dimensionality Reduction 및 예측 모델링 과정에서 입력 피처로 활용되어, 텍스트 정보가 금융 시장 예측 성능에 미치는 영향을 평가하는 데 사용되었다.

## 2.3 Feature Engineering

본 프로젝트에서는 수집된 뉴스 텍스트 임베딩 벡터와 경제 지표 데이터를 결합하여, 시장 변동성과 관련된 다양한 입력 feature 를 구성하였다. 각 feature set은 텍스트 정보의 포함 여부와 보조 지표의 활용 여부에 따라 순차적으로 확장되며, 최종적으로 총 4가지 feature 구성안(A-D) 을 설계하였다.

### A. Baseline features

- 구성: [date\_index] + [S&P500 시계열 데이터]
- S&P500 시계열 데이터는 시점 간 관계를 반영하기 위해 최대 5일(lag 5)까지의 지연값을 포함하였다. 이 구성은 텍스트 정보를 배제한 순수 시계열 기반 baseline으로, 다른 복합 피처 조합과 비교했을 때 순수 시계열 모델의 예측 성능을 평가하기 위한 기준점으로 사용되었다.

### B. Text-Augmented features

- 구성: [date\_index] + [S&P500(lag5)] + [text embedding]

- A 구성에 텍스트 임베딩 벡터를 추가하여, 뉴스의 의미 정보를 반영한 형태이다. 각 기사에서 얻은 임베딩 벡터는 해당 날짜의 시장 데이터와 매칭되도록 처리하였다. 이를 통해 뉴스의 의미적 맥락이 단기 시장 변동에 미치는 영향을 반영할 수 있도록 설계되었다.

#### C. Person-conditioned features

- 구성: [date\_index] + [S&P500(lag5)] + [text embedding] + [person one-hot encoding]
- B 구성에 인물 정보를 one-hot encoding 형태로 추가하였다. 모델이 인물별 발언이나 보도 내용이 시장 변동에 미치는 차이를 학습할 수 있도록 하였으며, 특정 인물 관련 뉴스에 대한 시장의 반응 차이를 분석하는데 사용되었다.

#### D. Extended features with Fear&Greed index

- 구성: [date\_index] + [S&P500(lag5)] + [text embedding] + [person one-hot encoding] + [fear & greed index]
- C 구성에 추가로 Fear & Greed Index를 결합하였다. 해당 지표는 시장 참여자의 심리적 요인을 정량화한 데이터로, 공포(Fear)와 탐욕(Greed) 수준을 통해 시장의 위험 선호도를 나타낸다. 이를 추가함으로써 단순 시계열 패턴뿐 아니라 시장 심리 요인이 예측 성능에 미치는 영향을 함께 검증할 수 있었다.

## 2.4 Dimensionality Reduction

텍스트 임베딩을 통해 생성된 벡터는 차원이 높고(최대 1024차원), 피처 수가 많아질수록 모델의 복잡도가 증가하여 overfitting이 발생할 가능성이 있다. 따라서 본 프로젝트에서는 모델의 일반화 성능을 유지하면서도 핵심 정보만을 보존하기 위해 두 가지 차원 설정 방식을 비교하였다.

#### X. Original dimension

- 모든 피처 벡터를 차원 축소 없이 그대로 사용하였다.
- 임베딩 모델이 학습한 의미적 표현을 손실 없이 반영할 수 있다는 장점이 있으며, baseline으로 사용하였다.

#### Y. PCA-based reduction

- Principal Component Analysis (PCA)[\[5\]](#)를 적용하여 차원을 축소하였다.
- Proportion of Variance, PoV 가 0.9가 될 때까지 주성분을 선택하여, 전체 변동성의 90%를 유지하도록 구성하였다. 이를 통해 불필요한 중복 차원을 제거하면서도 텍스트의 핵심 의미를 최대한 보존하였다.

이 두 가지 설정은 이후 실험 단계에서 각각의 feature 조합(A-D) 및 모델(linear regression, SARIMA, LightGBM, GRU)에 적용되어, 차원 축소 여부가 예측 성능에 미치는 영향을 비교·분석하는 데 사용되었다.

## 2.5 Model Selection

뉴스 텍스트 임베딩과 경제 지표 데이터를 활용하여 S&P500 변동을 예측하기 위해, 본 프로젝트는 네 가지 유형의 모델을 비교하였다: (1) 전통적 통계 회귀모델, (2) 시계열 기반 통계모델, (3) 머신러닝 모델, (4) 딥러닝 모델

각 모델은 데이터의 구조적 특성과 복잡도에 따라 상호보완적인 역할을 수행하도록 설계되었으며, 단순 선형 구조부터 비선형·시계열 구조까지 점진적으로 복잡도를 높이는 형태로 구성되었다.

#### A. Linear Regression[\[6\]](#)

Linear Regression은 입력 feature와 출력값 간의 선형 관계를 학습하는 대표적 통계 회귀모델이다. 구조가 단순해 변수 간 영향력을 직접 해석할 수 있고, 경제·금융 예측에서 가장 널리 사용되는 baseline 모델로 활용된다. 모든

feature를 가중합 형태로 결합해 예측값을 계산하며, 회귀 계수의 부호와 크기로 변수의 기여도를 해석할 수 있다. 과적합에 강하고 학습 속도가 빨라, 복잡한 모델과의 성능 비교 기준으로 적합하다.

## B. SARIMA[\[7\]](#)

SARIMA (Seasonal AutoRegressive Integrated Moving Average)는 시계열 데이터의 자기상관성과 계절적 주기를 함께 고려하는 대표적 통계모델이다. AR(자기회귀), I(차분), MA(이동평균) 구조에 Seasonal 항(S) 을 더해 주기적 변동을 반영하며, 시간적 패턴이 중요한 금융 데이터 분석에 적합하다. 본 프로젝트에서는 전통적 시계열 예측의 baseline 모델로 사용되어, 이후 머신러닝·딥러닝 모델의 확장 성능 비교 기준으로 활용되었다.

## C. LightGBM[\[8\]](#)

LightGBM(Light Gradient Boosting Machine)은 트리 기반 머신러닝 모델의 대표적 구현체로, 비선형 관계와 feature 간 상호작용을 효율적으로 학습한다. Gradient Boosting 방식으로 여러 weak learner들을 단계적으로 결합하여 예측 오차를 줄이며, 학습 속도와 메모리 효율이 높다. 또한 자동 feature selection, 결측치 처리, feature importance 산출 기능을 내장하고 있어 고차원 데이터 분석에 적합하다. 본 모델은 머신러닝 접근법의 대표로서, 복잡한 비선형 패턴을 포착할 수 있다는 점에서 선형 회귀나 전통적 시계열 모델과 구분된다. 따라서 LightGBM은 비선형 관계를 기반으로 한 금융 예측 모델의 대표적 사례로 설정되었다.

## D. GRU[\[9\]](#)

GRU(Gated Recurrent Unit)는 딥러닝 기반 시계열 모델의 대표 구조로, RNN의 한 변형이다. GRU는 update gate와 reset gate 두 가지 게이트를 통해 시간적 정보의 흐름을 제어하며, 장기 의존성과 단기 변화를 동시에 학습할 수 있다. 이는 금융 시계열 데이터처럼 시점 간 상호영향이 누적되는 데이터에 적합하다. LSTM(Long Short-Term Memory)보다 구조가 간결하고 학습 효율이 높아, 시계열 텍스트 임베딩 및 연속적 시장 반응 모델링에서 자주 사용된다. GRU는 본 프로젝트에서 deep learning 기반의 sequential learning 접근을 대표하는 모델로, 시간의 흐름 속에서 뉴스 의미가 누적되어 시장 반응으로 전이되는 과정을 학습하도록 설계되었다.

이 네 가지 모델은 각각 선형적, 통계적, 비선형적, 그리고 시계열 딥러닝적 접근 방식을 대표하며, 동일한 feature 조합과 차원 설정 하에서 성능을 비교함으로써 텍스트 정보가 금융시장 예측에 기여하는 정도를 다각적으로 평가하였다.

## 2.6 Experiment setup

모델 학습은 2017년부터 2019년까지의 데이터를 활용하여 진행하였다. 2017-2018년 데이터를 train set으로, 2019년 초반 데이터를 validation set으로, 그리고 2019년 후반 데이터를 테스트 test set으로 사용하였다. 이러한 시계열 분할은 데이터의 시간적 순서를 보존하면서도, 과거 뉴스와 시장 지표를 기반으로 미래 변동을 예측하는 실제 금융 예측 환경을 모사하기 위함이다.

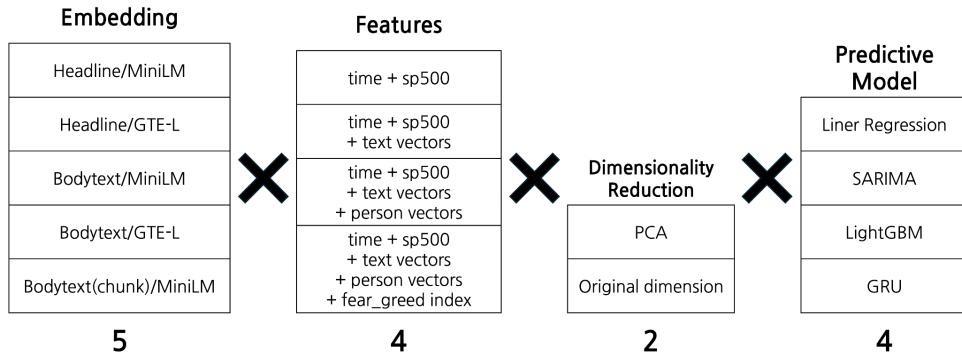
Loss function은 Mean Squared Error(MSE)를 기본으로 사용하였다. 그러나 모델 간 비교의 직관성을 높이기 위해 Root Mean Squared Error(RMSE)와 R-square(결정 계수) 역시 함께 계산하여 예측 성능을 다각도로 평가하였다. RMSE는 예측 오차의 크기를, R-square는 모델이 실제 데이터 변동을 얼마나 설명하는지를 나타내므로, 두 지표의 병행 사용을 통해 모델의 정밀도와 설명력을 동시에 검증하였다.

실험은 임베딩(Embedding) × 피처(Feature) × 차원 축소(Dimension) × 모델(Model) 의 네 가지 요소를 조합하여 수행하였다. 각 조합의 구성은 다음과 같다.

- **Embedding:** 5가지 방식

- **Feature:** 4가지 구성 (A~D)
- **Dimension:** 2가지 설정 (Full(X) / PCA(Y))
- **Model:** 4가지 유형 (Linear Regression, SARIMA, LightGBM, GRU)

총 조합수는  $5 \times 4 \times 2 \times 4 = 160$  으로, 총 160가지 실험 구성을 학습 및 평가하였다. 모든 실험은 동일한 데이터 분할, 평가지표, 그리고 하이퍼파라미터 세팅 하에서 수행되었으며, 이를 통해 각 구성 요소가 예측 성능에 미치는 영향을 체계적으로 비교·분석하였다. Figure 1은 본 프로젝트의 전체 실험 조합 구조를 시각적으로 요약한 것이다.



[Figure 1] 임베딩, feature, 차원 축소, 예측 모델의 조합에 따른 실험 설계 개요

### 3. Results and Analysis

- **Column 설명**
  - feature\_type: **A**(time+sp500) / **B**(A+text vectors) / **C**(B+person vectors) / **D**(C+fear\_greed index)
  - dim\_type: **X**(Original dimension) / **Y**(PCA)

#### 1. Performance comparison across models

실험 결과, Linear Regression 모델과 SARIMA 모델이 유의미한 예측력을 보인 반면, GRU나 LightGBM 같은 복잡한 모델은 모두  $R^2$ 가 음수를 기록하며 예측에 실패했다. 특히 주목할 점은, Baseline feature만을 사용한 **Linear Regression 모델이  $R^2$  0.94로, 모든 실험 중 가장 높은 성능을 달성했다는 것이다.** 이는 S&P500 지수의 변동성 중 94%를 설명할 수 있는 매우 강력한 성능이며, 금융 시계열 데이터의 강한 autocorrelation 특성을 명확히 보여준다.

embedding	feature_type	dim_type	model	MSE	RMSE	R2
miniLM_headline_outputs	A	X	LinearRegression	558.980221728731	23.6427625655026	0.941482428300521
miniLM_headline_outputs	A	X	SARIMA	692.52896272732	26.3159450282014	0.927501704613033
miniLM_headline_outputs	B	X	LinearRegression	2856.96426082378	53.4505777407858	0.700914979677515
miniLM_headline_outputs	B	X	SARIMA	2900.04448552006	53.8520611074456	0.696405070311317
miniLM_headline_outputs	A	X	GRU	14430.2687725498	120.126053679249	-0.510651459060439
miniLM_headline_outputs	C	X	LinearRegression	20666.4690527574	143.758370374588	-1.16349619818335
miniLM_headline_outputs	C	X	SARIMA	20719.708541472	143.943421320573	-1.16906964331967
miniLM_headline_outputs	B	X	LightGBM	26103.7814463851	161.566647072918	-1.73270831961146
miniLM_headline_outputs	C	X	LightGBM	26293.1735193615	162.151699094895	-1.75253507515468
miniLM_headline_outputs	D	X	LightGBM	26473.3979132569	162.706477785172	-1.7714021002869

#### 2. Effect of Model Complexity and Data Noise

금융 데이터는 시장 참여자의 감정, 루머, 무작위적 거래 등으로 인해 noise가 많은 특성이 있다. GRU와 같은 복잡한 모델은 이러한 noise에 쉽게 overfitting되는 반면, Linear Regression 같은 단순한 모델은 가장 강력하고 일관된

선형 관계만을 추출하여 안정적인 성과를 낸 것으로 해석된다. 이러한 분석은 텍스트 임베딩 feature를 추가할수록  $R^2$  값이 하락한 결과와도 일치하며, **추가된 feature가 시그널이 아닌 noise로 작용했음을 시사한다.**

embedding	feature_type	dim_type	model	MSE	RMSE	R2
miniLM_headline_outputs	A	X	GRU	14430.2687725498	120.126053679249	-0.510651459060439
miniLM_headline_outputs	B	X	GRU	44444.0957290569	210.81768362511	-3.65268797955023
miniLM_headline_outputs	D	X	GRU	57505.4384001904	239.802915745806	-5.02003162882237
miniLM_headline_outputs	C	X	GRU	57937.70145552	240.702516512644	-5.06528364910863
miniLM_headline_outputs	A	X	LinearRegression	558.980221728731	23.6427625655026	0.941482428300521
miniLM_headline_outputs	B	X	LinearRegression	2856.96426082378	53.4505777407858	0.700914979677515
miniLM_headline_outputs	C	X	LinearRegression	20666.4690527574	143.758370374588	-1.16349619818335
miniLM_headline_outputs	D	X	LinearRegression	45154.2838941467	212.495373818224	-3.7270349515097

### 3. Impact of PCA on text embedding performance

유의미한 예측 성능을 보인 Linear Regression과 SARIMA 모델에서, text embedding 원본보다 PCA를 적용했을 때  $R^2$  값이 예외 없이 상승했다. 특히 Linear Regression 모델과 SARIMA 모델 모두  $R^2$ 가 0.70에서 0.93으로 급등한 사례는, PCA가 고차원 텍스트 벡터의 noise를 제거하고 유의미한 정보만 추출했음을 명확히 보여준다. 이는 **고차원의 텍스트 데이터에 noise가 많이 포함된 경우, PCA와 같은 차원 축소 기법이 유의미한 정보 추출을 위한 필수적인 전처리 과정일 수 있음을 시사한다.** 나아가, 이는 뉴스 기사 텍스트 안에 S&P500 지수의 변동성을  $R^2$  0.93 수준으로 설명할 수 있는 강력한 신호가 존재하며, PCA를 통해 noise를 잘 제거한다면, 텍스트 기반 예측 접근법이 충분히 실효성이 있음을 보여준다.

embedding	feature_type	dim_type	model	MSE	RMSE	R2
gte_large_headline_outputs	B	Y	LinearRegression	629.7559888684	25.0949395071675	0.934073175043988
gte_large_headline_outputs	B	X	LinearRegression	2813.07135526811	53.0383951045666	0.705509965596711
gte_large_headline_outputs	B	Y	SARIMA	631.061516539334	25.1209378117007	0.933936504181371
gte_large_headline_outputs	B	X	SARIMA	2857.02234074031	53.4511210428772	0.700908899505896

### 4. Effect of Additional Features (Person and Fear & Greed Index)

text embedding을 추가한 이후, person one-hot encoding 및 fear & greed index feature들을 추가했을 때, 모델별로 상반된 결과가 나타났다. Linear Regression 모델에서는 feature들이 추가될수록 성능이 하락하는 경향을 보인 반면, SARIMA 모델에서는  $R^2$ 가 점진적으로 상승하는 긍정적인 효과를 확인할 수 있다. 이는 두 가지 가능성을 시사한다. 첫째는 **해당 feature 자체가 예측에 방해가 되었거나, 혹은 단순 추가하는 정보 전달하는 방식이 비효율적이었을 가능성이다.** 둘째는 이 feature들이 유의미한 신호를 포함하고 있었으나, Linear Regression과 같은 모델이 이 데이터의 신호를 효율적으로 해석하지 못했을 수도 있다. 따라서 **향후 더 정교한 feature engineering을 통해 해당 feature들을 text embedding과 결합한다면, SARIMA 모델에서 확인한 긍정적 효과를 극대화할 수 있을 것으로 기대된다.**

embedding	feature_type	dim_type	model	MSE	RMSE	R2
miniLM_headline_outputs	B	Y	LinearRegression	642.177215358595	25.3412157434996	0.932772842789856
miniLM_headline_outputs	D	Y	LinearRegression	847.979666411965	29.1200904258892	0.911228145469083
miniLM_headline_outputs	C	Y	LinearRegression	854.03601052964	29.2238945133881	0.910594129206313
miniLM_headline_outputs	D	Y	SARIMA	849.985677716144	29.15451384805	0.911018143566048
miniLM_headline_outputs	C	Y	SARIMA	1092.15188970486	33.0477213995891	0.885666658625462
miniLM_headline_outputs	B	Y	SARIMA	1319.59572164146	36.3262401253069	0.861856405193252

### 5. Comparison of Embedding Strategies (Headline vs. Bodytext vs. Chunking)

성공적인 모델(Linear Regression, SARIMA)를 기준으로 임베딩 방식을 비교한 결과, 전반적으로 headline보다 bodytext를 사용했을 때 근소하게 더 우수한 성능을 보이는 경향이 나타났다. 이는 headline이 요약적 정보에 집중되어 있지만, **bodytext는 문맥적·구체적 정보를 포함하여 더 안정적인 예측 신호를 제공했기 때문으로 해석된다.**

반면, chunking 방식은 주로 headline이나 bodytext 방식보다 더 낮은  $R^2$  값을 기록했다. 본문을 일정 단위로 분할하고 overlap을 적용하는 과정에서, 기사의 핵심 의미가 분산되거나 문맥이 단절되어 노이즈가 증가했을 가능성이 있다. 이러한 결과는, 금융 뉴스 텍스트의 의미적 신호는 국소적 단위보다는 전체 문맥 수준에서 더 잘 포착될 수 있음을 시사한다.

embedding	feature_type	dim_type	model	MSE	RMSE	R2
miniLM_chunking_outputs	B	Y	SARIMA	935.038413716296	30.5783978278178	0.90211428725118
miniLM_bodytext_outputs	B	Y	SARIMA	1021.84852265689	31.9663654902601	0.893026458063841
miniLM_headline_outputs	B	Y	SARIMA	1319.59572164146	36.3262401253069	0.861856405193252
miniLM_bodytext_outputs	C	Y	SARIMA	951.948143943647	30.8536568974189	0.900344070144146
miniLM_headline_outputs	C	Y	SARIMA	1092.15188970486	33.0477213995891	0.885666658625462
miniLM_chunking_outputs	C	Y	SARIMA	1530.00336704524	39.1152574712891	0.839829607110923
miniLM_headline_outputs	D	Y	SARIMA	849.985677716144	29.15451384805	0.911018143566048
miniLM_bodytext_outputs	D	Y	SARIMA	1012.60986882989	31.8215315286661	0.893993618558458
miniLM_chunking_outputs	D	Y	SARIMA	1589.19811666808	39.8647477938602	0.833632727739101

## 4. Limitations and Future Improvements

### 4.1 Applying Clustering for Data Characterization

- 개선 필요성:** 현재는 모든 뉴스 기사 데이터를 동일한 가중치로 모델이 입력하고 있다. 그러나 기사의 주제나 논조에 따라 시장에 미치는 영향력이 다를 수 있다.
- 개선 계획:** 데이터의 숨겨진 특성을 파악하기 위해 k-means과 같은 클러스터링 알고리즘을 도입한다. gte, miniLM 등으로 추출한 기사 임베딩 벡터를 기반으로 클러스터링을 수행하여, 유사한 주제(e.g., 금리 인상, 무역 분쟁, 신기술 발표)를 가진 기사들을 그룹화할 것이다. 이후, 특정 클러스터의 기사가 등장했을 때 주가 변동성이 통계적으로 유의미하게 달라지는지 분석하여, 이를 새로운 feature로 활용할 계획이다.

### 4.2 Advanced Feature Engineering

- 개선 필요성:** 현재 headline과 body 전체를 임베딩 하는 방식은 기사의 핵심 의미를 정확히 포착하지 못하고 불필요한 noise를 포함할 수 있다. 또한, 텍스트 외의 시장 상황을 반영할 추가 정보가 부족하다.
- 개선 계획**
  - 임베딩 전략 수정:** 뉴스 기사가 대부분 두괄식 구성이거나 결론이 마지막에 나오는 특성을 고려하여, 텍스트 전체가 아닌 ‘첫 번째 문단’과 ‘마지막 문단’의 텍스트만을 추출한다. 이후, 문장 간의 의미론적 관계 포착에 강점이 있는 SBERT 모델을 활용해 임베딩을 시도하여, 더 압축적이고 핵심적인 시그널을 포착할 것이다.
  - 추가 지표 탐색:** 뉴스 기사 텍스트 외에 예측력을 높일 수 있는 다른 지표를 적극적으로 탐색한다. 예를 들어, VIX 지수와 같은 시장의 공포 및 심리 지표, 또는 주요국 금리나 유가(WTI) 데이터를 추가 feature로 결합하여, 모델이 시장의 복합적인 맥락을 함께 학습하도록 개선한다.

### 4.3 Model Refinement and Deep Learning Enhancement

- 개선 필요성:** LightGBM과 GRU에서 텍스트 feature의 가능성을 확인했으나, 각 모델의 잠재력을 극대화하기 위한 최적화 과정이 부족했다.
- 개선 계획:** 기존 모델에 대한 집중적인 튜닝과 함께, 더 복잡한 시계열 딥러닝 모델을 도입한다. GRU 모델의 하이퍼파라미터 튜닝을 진행하고, LSTM 등 대안적인 RNN 모델을 적용하여 성능을 비교한다. 나아가 텍스트와 시계열 데이터를 보다 정교하게 결합할 수 있는 Attention 메커니즘을 도입하거나, Transformer 기반의 시계열 예측 아키텍처를 구현하여 예측 성능의 한계를 시험해볼 계획이다.

## References

- [1] TIME Magazine, “The 100 Most Influential People of 2018,” 2018. [Online]. Available: <https://time.com/collection/most-influential-people-2018/>
- [2] *The Guardian Open Platform*, “Search API Documentation,” 2024. [Online]. Available: <https://open-platform.theguardian.com/documentation/search>
- [3] W. Wang, B. Bi, M. Yan, C. Wu, Z. Bao, L. Peng, G. Si, et al., “MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers,” *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 5776-5788, 2020.
- [4] Alibaba-NLP, “GTE-large-en-v1.5,” *Hugging Face Repository*, 2024. [Online]. Available: <https://huggingface.co/Alibaba-NLP/gte-large-en-v1.5>
- [5] K. Pearson, “On lines and planes of closest fit to systems of points in space,” *Philosophical Magazine*, vol. 2, no. 11, pp. 559-572, 1901.
- [6] F. Galton, “The conception, history, and future of regression,” *Philosophical Transactions of the Royal Society of London. Series A*, vol. 187, pp. 71-89, 1896.
- [7] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. San Francisco, CA: Holden-Day, 1970.
- [8] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, “LightGBM: A highly efficient gradient boosting decision tree,” *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
- [9] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724-1734, 2014.