[기존 접근]
'코스피'라는 키워드를 이용하여 네이버 기사를 추출하고 데이터 하나하나 일일이 레이블링하는 작업을 거침

[feed back]
주식보다 Y값을 주가로 상정하여 예측한다고 가정한다.

1. 원천데이터의 잘못 => 매일 경제 타이들을 다 집어넣고 다음날 주가를 예측 하는 방안
- 사람은 가이드를 만들 뿐이고 컴퓨터 스스로 패턴으로 라벨링을 하도록 만들 것
- 키워드 설정 자체를 잘못했다.

2. 자연어 처리 스텝 자체를 바꿀 것
- 도메인 지식 자체가 부족한 상태에서 현재 접근 방법은 실제 주가 예측하기가 어렵고 허황된 방법에 가깝다.
- 전체 기사를 수집하고 전체를 1 혹은 0으로 부여하여 컴퓨터가 패턴을 분석하도록 하는 방안이 시간적인 면에서도, 정확도 측면에서도 더 우월할 것

3. Vader 모형
- 극성 정리 : 명확하게 긍정적, 부정적인 것만 사용해야 함
=> 코스피 정리가 어려울 수 있음
- 불용어 처리, 필요없는 문장 배제 => 0 이나 1로 넣을 것
- 결론적으로 Compound의 정확도가 높아야 할 것
=> 눈으로 보는 정확도 <=> 알고리즘의 결과 라벨링과 일치 되어야 할 것

4. LDA
- LDA 성능지표에 따라 몇 개 그룹이 적합? => 토픽 개수 도출(coherence, perplexity => k값 도출)
- 완성하는데 좀 오래 걸림

★ Colab으로 정리해서 모두 활용할 수 있게끔 만들 것! 차트까지 나오도록

[how?]
1. ① 데이터 몇 개? ② 어떤 시점? ③ 어느정도 등락폭을 기준? (5%? vs 2%?)
=> 자체 합의가 필요. 어느정도 베이스 라인이 생기면 추후 수정하면서 작업이 가능할 것

2. 모든 기사를 수집할 것이 아니라, 카테고리나 topic을 정하는 방법 등을 취하면서 접근할 것 => 크롤링 코드 자체를 수정하는 방안이 필요된다고 보임.

3. 기사 컨텐츠는 불용어 처리와 같이 불완전한 요소가 많으므로 기본 접근은 기사 제목만 활용하는 기존의 방법을 취할 것

4. 코스피에 영향을 주는 단어를 파악하는 것이 관건이 될 것

[목표]
- 차주는 성능 개선 차원(베이스 라인)으로 넘어가야 할 것 => ★기본 성능 나와야
- 시간의 60퍼센트는 성능 개선의 문제에 쏟아야 할 것