LDA

데이터 전처리

NLTK(Natural Language Toolkit) 기반으로 수행.
: 자연어 처리 라이브러리

1. 소문자 변환 : 텍스트 전부 소문자화
2. 공백 및 개행 제거 : 불필요한 공백과 개행 제거
3. 토큰화 : `word_tokenize`를 사용하여 텍스트를 단어 단위로 분리
4. 알파벳 이외 제거 : 숫자, 특수 문제 토큰 제거
5. 불용어 제거 : NLTK의 불용어 사전 통해 기본 제거 + 추가적으로(내가) ('may', 'one', first', 'second', 'third') 눈에 보이는 이상한 애들만 조금 제거
6. 품사 태깅: 표제어 추출할려고 명사, 동사, 형용사, 부사 구분
7. 레마타이제이션(표제어 기본형으로 변환): `WordNetLemmatizer`통해서 기본형으로 변환

Grid Search를 이용한 파라미터(Passes, Topics) 선정

: 최적 해 찾으려고 변수 2개니깐 이중 for문 했다고 생각하면 쉽습니다.

파라미터 설명
Topics: 돌려서 토픽 몇 개 뽑아낼건지, BERTopic에서 260개 나왔던 것처럼
Passes : 데이터셋 전체를 몇 번 반복 처리할 지, 데이터를 여러 번 통과 시켜서 파라미터를 업데이트 시켜주는 것. (마냥 많이 한다고 좋은 건 아님, 과적합 위험)

평가지표 설명
Coherence : 일관성. 도출한 토픽이 얼마나 일관성 있는지, 높으면 토픽 내의 단어들이 일관되고, 해석하기 쉽고 잘 어울린다는 뜻 (그렇다고 너무 높으면 정보양 줄어듦)
Perplexity : 혼란도. 작을수록 토픽모델이 문서를 잘 반영하는 것. 즉, 특정 확률 모델이 실제로 관측되는 값을 얼마나 잘 예측하는지.

Grid Search 결과
topic_range = [10, 15, 20, 25, 30]
passes_range = [10, 15, 20, 25, 30, 35, 40, 45, 50]
이게 한 사이클 돌릴때마다 데이터셋을 수십번 학습시키는거라
시간이 오래걸리는 관계로 위의 범위(5단위)만 돌렸음.

Topics: 10, Passes: 10, Coherence Score: 0.39212070937177357, Perplexity: -6.738197856463975
Topics: 10, Passes: 15, Coherence Score: 0.41938438620747853, Perplexity: -6.726627416482278
Topics: 10, Passes: 20, Coherence Score: 0.4138647123078399, Perplexity: -6.7245207190591465
Topics: 10, Passes: 25, Coherence Score: 0.4275313596790545, Perplexity: -6.7206442785213785
Topics: 10, Passes: 30, Coherence Score: 0.43146421991012335, Perplexity: -6.710802946706033
Topics: 10, Passes: 35, Coherence Score: 0.44479984210895473, Perplexity: -6.7044231545835276
Topics: 10, Passes: 40, Coherence Score: 0.4248084921125269, Perplexity: -6.708019260115556
Topics: 10, Passes: 45, Coherence Score: 0.4522310853447237, Perplexity: -6.704735299344614
Topics: 10, Passes: 50, Coherence Score: 0.4587133534096042, Perplexity: -6.709686798844093 <- 데이터 정제 끝나면 다시 해야겠지만 우선 얘가 최고.
Topics: 15, Passes: 10, Coherence Score: 0.408008832801028, Perplexity: -6.899726292309499
Topics: 15, Passes: 15, Coherence Score: 0.4092241053805836, Perplexity: -6.869302443808976
Topics: 15, Passes: 20, Coherence Score: 0.41074060975265775, Perplexity: -6.887479746954163
Topics: 15, Passes: 25, Coherence Score: 0.4124938838923787, Perplexity: -6.868318832160241
Topics: 15, Passes: 30, Coherence Score: 0.42670577844896423, Perplexity: -6.8692300945073494
Topics: 15, Passes: 35, Coherence Score: 0.41146903082082525, Perplexity: -6.887866187508984
Topics: 15, Passes: 40, Coherence Score: 0.41431269706184504, Perplexity: -6.859654202704622
Topics: 15, Passes: 45, Coherence Score: 0.4143662077114268, Perplexity: -6.883565925599826
Topics: 15, Passes: 50, Coherence Score: 0.4251189743094182, Perplexity: -6.874871950983334
Topics: 20, Passes: 10, Coherence Score: 0.40329035478777014, Perplexity: -6.953497840192233
Topics: 20, Passes: 15, Coherence Score: 0.3945085860700001, Perplexity: -6.954727974595018
Topics: 20, Passes: 20, Coherence Score: 0.4045435181033559, Perplexity: -6.940391233270686
.

.
.

그 외 파라미터

1. `no_below=5` : 5개 미만의 문서에서 등장하는 단어들은 제거
2. `no_above=0.5` : 전체 문서의 50% 이하에서만 등장하는 단어만 포함 (엄청 빈번하게 포함된 단어는 빠졌다는 뜻.)
3. `iterations=400` : 각 문서의 주제 할당을 400번 반복하면서 업데이트 (passes랑 비슷해보이는데 조금 다름, 얘는 주제 할당을 정교하게, passes는 데이서셋을 기준으로 모델이 얼마나 학습할 건지(일반화할건지))

그래서 나온 토픽 결과

(0, '0.031*"supply"* + *0.030*"chain" + 0.024*"entity"* + *0.023*"asset" + 0.022*"product"* + *0.018*"digital" + 0.017*"distribution"* + *0.016*"use" + 0.013*"provide"* + *0.013*"invention"')
(1, '0.111*"container"* + *0.023*"cargo" + 0.017*"storage"* + *0.013*"position" + 0.013*"device"* + *0.012*"load" + 0.011*"ship"* + *0.011*"member" + 0.010*"sensor"* + *0.010*"temperature"')
(2, '0.035*"information"* + *0.031*"item" + 0.026*"ship"* + *0.023*"location" + 0.023*"method"* + *0.018*"order" + 0.018*"delivery"* + *0.017*"user" + 0.016*"service"* + *0.016*"receive"')
(3, '0.039*"module"* + *0.038*"power" + 0.038*"control"* + *0.031*"device" + 0.029*"configure"* + *0.019*"couple" + 0.019*"port"* + *0.018*"unit" + 0.018*"controller"* + *0.018*"circuit"')
(4, '0.108*"data"* + *0.017*"set" + 0.017*"base"* + *0.016*"plurality" + 0.016*"method"* + *0.014*"least" + 0.014*"process"* + *0.013*"generate" + 0.013*"determine"* + *0.013*"model"')
(5, '0.045*"signal"* + *0.023*"image" + 0.022*"object"* + *0.021*"sensor" + 0.014*"least"* + *0.012*"tag" + 0.012*"use"* + *0.011*"device" + 0.011*"information"* + *0.010*"receive"')
(6, '0.070*"network"* + *0.058*"device" + 0.023*"user"* + *0.018*"communication" + 0.016*"access"* + *0.015*"service" + 0.015*"provide"* + *0.014*"server" + 0.012*"application"* + *0.012*"use"')
(7, '0.072*"vessel"* + *0.023*"fluid" + 0.015*"water"* + *0.015*"gas" + 0.013*"flow"* + *0.013*"pressure" + 0.012*"liquid"* + *0.012*"sample" + 0.011*"heat"* + *0.011*"material"')
(8, '0.041*"node"* + *0.032*"communication" + 0.029*"information"* + *0.028*"device" + 0.026*"wireless"* + *0.025*"network" + 0.021*"terminal"* + *0.020*"base" + 0.019*"receive"* + *0.019*"method"')
(9, '0.073*"vehicle"* + *0.036*"transport" + 0.034*"unit"* + *0.030*"control" + 0.015*"method"* + *0.014*"station" + 0.014*"transportation"* + *0.012*"provide" + 0.011*"operation"* + *0.010*"route"')

1. 토픽 0:
   - 공급(supply) 체인(chain) 엔티티(entity) 자산(asset) 제품(product) 디지털(digital) 분배(distribution) 사용(use) 제공(provide) 발명(invention)
2. 토픽 1:
   - 컨테이너(container) 화물(cargo) 저장(storage) 위치(position) 장치(device) 하중(load) 배(ship) 부품(member) 센서(sensor) 온도(temperature)
3. 토픽 2:
   - 정보(information) 품목(item) 배(ship) 위치(location) 방법(method) 주문(order) 배달(delivery) 사용자(user) 서비스(service) 수신(receive)
4. 토픽 3:
   - 모듈(module) 전력(power) 제어(control) 장치(device) 구성(configure) 결합(couple) 포트(port) 단위(unit) 컨트롤러(controller) 회로(circuit)
5. 토픽 4:
   - 데이터(data) 세트(set) 기반(base) 다수(plurality) 방법(method) 최소(least) 과정(process) 생성(generate) 결정(determine) 모델(model)
6. 토픽 5:
   - 신호(signal) 이미지(image) 객체(object) 센서(sensor) 최소(least) 태그(tag) 사용(use) 장치(device) 정보(information) 수신(receive)
7. 토픽 6:
   - 네트워크(network) 장치(device) 사용자(user) 통신(communication) 접근(access) 서비스(service) 제공(provide) 서버(server) 응용프로그램(application) 사용(use)
8. 토픽 7:
   - 용기(vessel) 유체(fluid) 물(water) 가스(gas) 흐름(flow) 압력(pressure) 액체(liquid) 샘플(sample) 열(heat) 재료(material)
9. 토픽 8:

- 노드(node) 통신(communication) 정보(information) 장치(device) 무선(wireless) 네트워크(network) 터미널(terminal) 기초(base) 수신(receive) 방법(method)

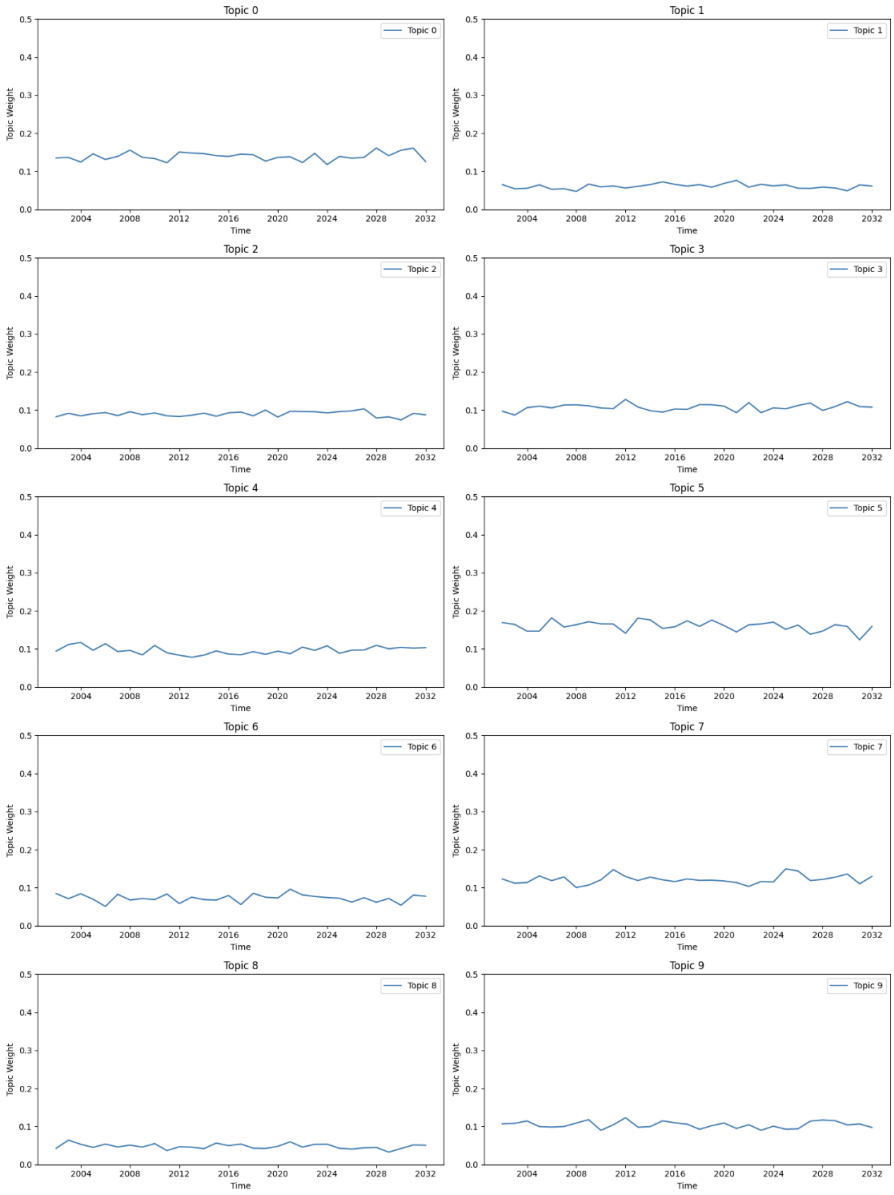
10. 토픽 9:

- 차량(vehicle) 운송(transport) 단위(unit) 제어(control) 방법(method) 역(station) 수송(transportation) 제공(provide) 운영(operation) 경로(route)

토픽 개수 10개라서 조금 포괄적인 단어들인데
토픽 개수 30개, 40개 늘리면 BERTopic 나왔던 것처럼 좀 세세한 단어들까지 나오긴함.

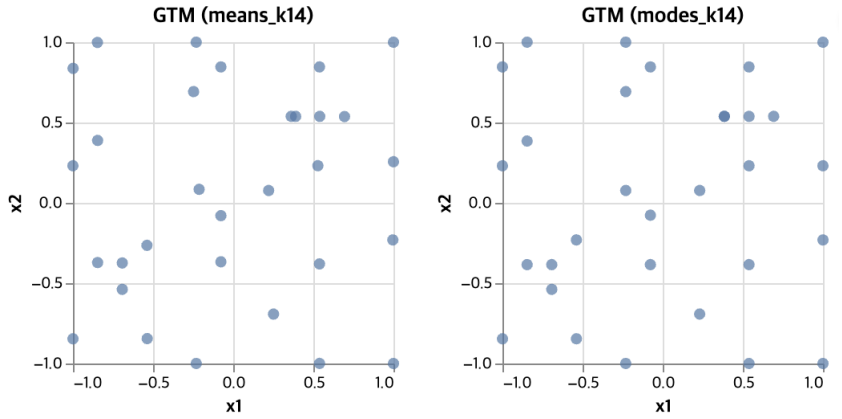
시계열 분석

위에서 LDA를 통해 얻은 토픽 분포에 날짜(공고일자)를 매치시켜서 연 단위로 어떻게 변화하는지 그래프로 시각화했음.
: 근데, 위에서 나온 토픽이 너무 일반적인 탓인지(?) 우상향하는 토픽이 1도 없고,
: 20년 동안 꾸준하게 나온 단어들임...
: 얘네 다 빼버려야되나싶음.
: 근데 위에서도 말했듯이 토픽 수 늘리면 조금 세세한 단어나오는데
: 그러고나서 돌려봐야할듯??
: 시계열 분석만 놓고봤을때는 잘나온 결과긴함
: 단어만 잘 골라지면 될듯.



GTM

: 마찬가지로 LDA로 얻은 토픽 분포를 교수님이 주신 코드에 때려박았음.



아직, GTM을 자세히 몰라서 그래프 해석이 힘든데
데이터의 고차원 구조를 저차원으로 투영하여 데이터의 구조를 시각화하는 방식이라고함.

그래서 데이터 분포를 시각화하는 건데, 데이터의 밀도와 패턴을 시각적으로 파악할 수 있음.
=> 이래서 공백기술이 파악가능한듯.
=> 이걸로 데이터의 분포, 클러스터링(얼마나 모여잇나?), 밀도, 중심 파악하는데

시계열 분석에서 봤듯이 뚜렷한 차이가 안보이는 듯한 느낌임

우선은 요기까쥐

BERTopic은 이것저것 뒤적거려봤는데 별다른 소득은 아직