

PortFolio_염성현
#Data Science
#ML/DL/NLP

CONTACT

kzvxy@naver.com

Education

2020.05. ~ 2020.11. 멀티 캠퍼스 / 인공지능 자연어처리(NLP) 기반 기업 데이터 분석 과정 수료

2012.02. ~ 2020.02. 한국외국어대학교 러시아학과 / 통계학과 졸업

2017.12. ~ 2018.01. 한국생산성본부 / 빅데이터를 활용한 SW 응용 분석 전문가 과정 수료

Skill

Language : Python, R, SAS, SQL

Tool : Pytorch, Tensorflow, Tableau

Projects

추천 : 자전거 정류장 최적 위치 선정 및 추천

개인화 : 감염병 유행 감지 및 개인 맞춤 정보 제공 서비스

NLP / DL : 논쟁적 에세이의 문장 요소 분류

Blog

<https://neulvo.tistory.com/category/Data>

PROJECT. 1

자전거 정류장 최적 위치 선정 및 추천

01

ABOUT PROJECT

최적 위치 선정, LSCP
위치 데이터 활용
folium 시각화
XGB with RFE
추천 알고리즘

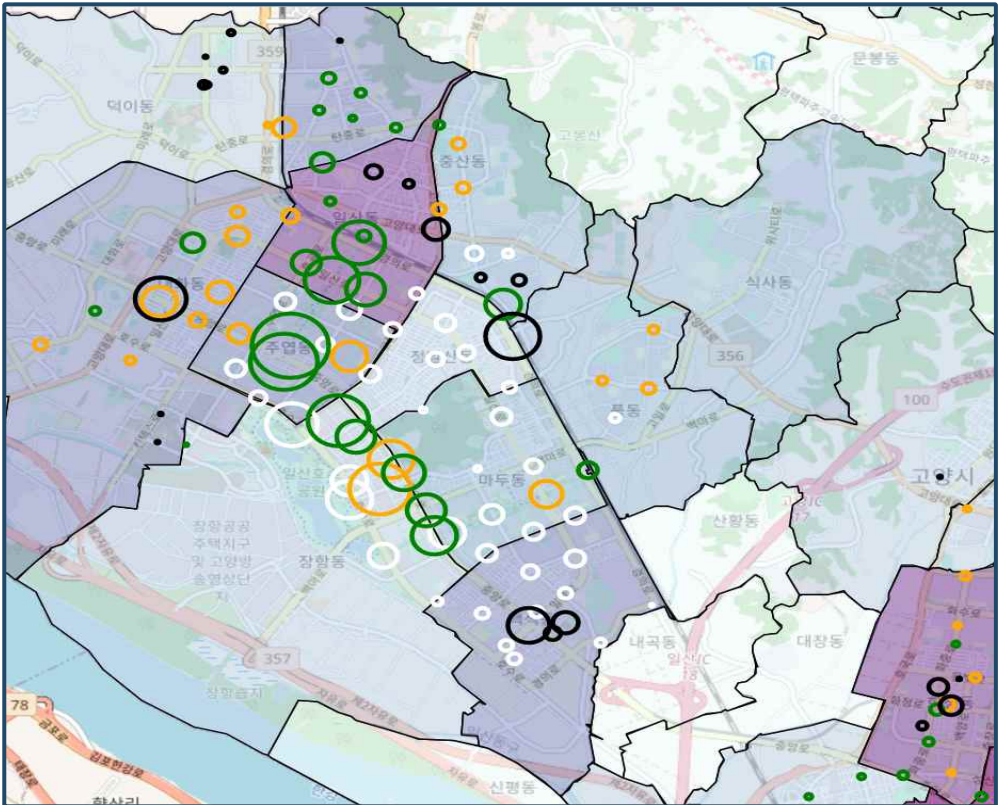
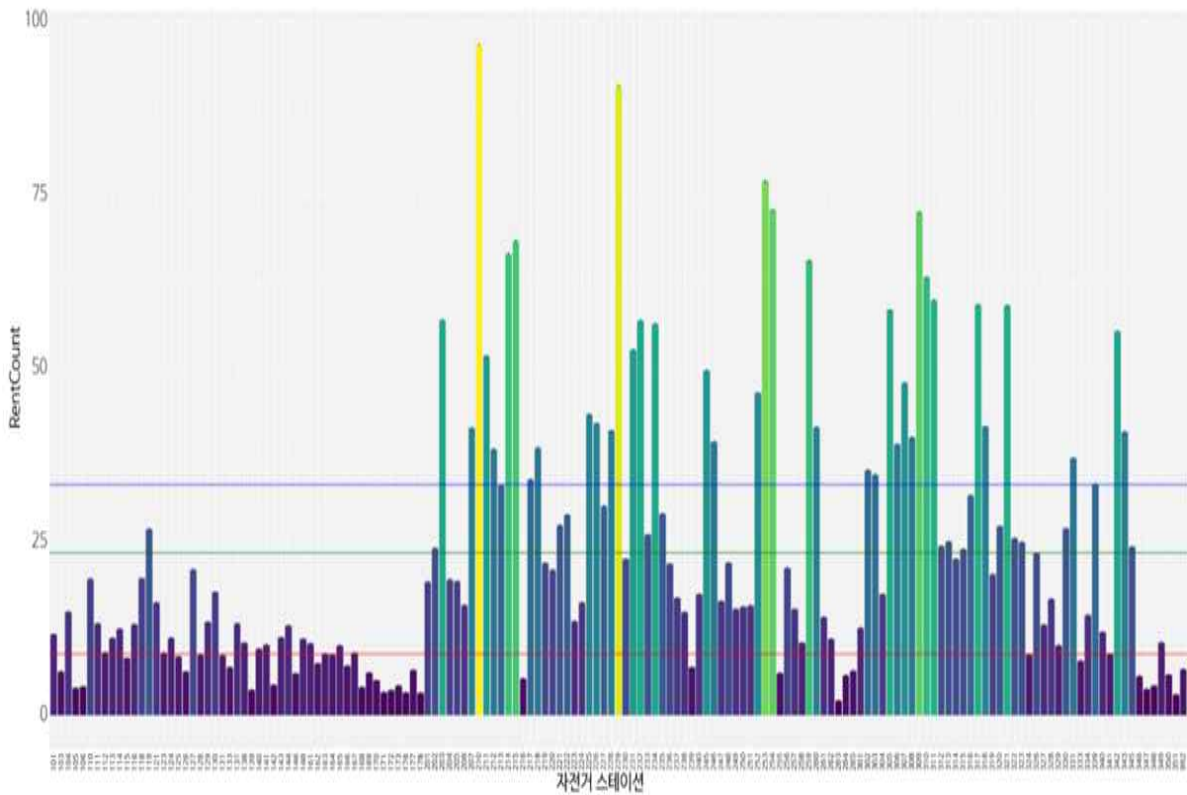
링크 :

<https://github.com/Neulvo/compass>

문제 정의

접근성 개선을 위한 공공 자전거 정류장의 최적 위치 선정 및 추천

EDA 및 문제 제기



정류장 별 이용량 격차 존재

- 정류장 별 이용량의 편차가 큰 것을 확인
- > 정류장 위치 조정을 통한 이용량 분담 및 전가 필요

* 2017 ~ 2019 정류장 별 총 대여 횟수를 시각화한 자료
빨간선 - 일사분위선, 초록선 - 중간값선, 파랑선 - 삼사분위선

자전거 분실 문제

- 정류장 별 상이한 수준의 자전거 분실 문제 발생
- > 자전거 분실 문제를 개선하는 정류장 위치 조정 필요

* 원의 크기 : 분실 후 관제 반납된 횟수
원의 색깔 : 대여 회수 비례 관제 반납 회수의 사사분위수
흰색, 노란색, 초록색, 검정색 순으로 1~4 사분위수
* 관제 반납 : 해당 정류장의 자전거를 관에서 직접 수거해간 것

신축 지구 내 정류장 미비

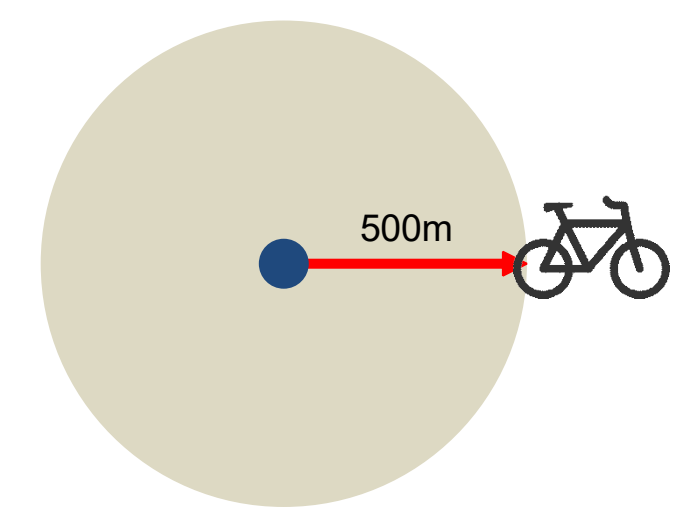
- 기존 정류장의 분포가 신축 지구의 수요를 반영하지 못함
- 이에 대한 커뮤니티 문의 사항 다수 존재
- > 신축 지구 수요를 반영한 정류장의 증설 필요

* 파란색 마커 : 기존 정류장 위치
빨간 원 : 신축 지구 위치

문제 해결

LSCP (Location Set Covering Problem)

- 영역 내 모든 수요를 커버하는 시설물의 최소 입지 수와 그 위치를 구하는 것
- 적정 거리의 영역을 설정하고 영역 안 기대 수요가 가장 높은 입지에 정류장을 위치시키기로 결정
- 거리 별 모델 설계 후, score가 가장 좋은 500m로 설정 (500m이상 거리 모델, score 하락 및 예측 수요 변동 미미)



기여 및 역할

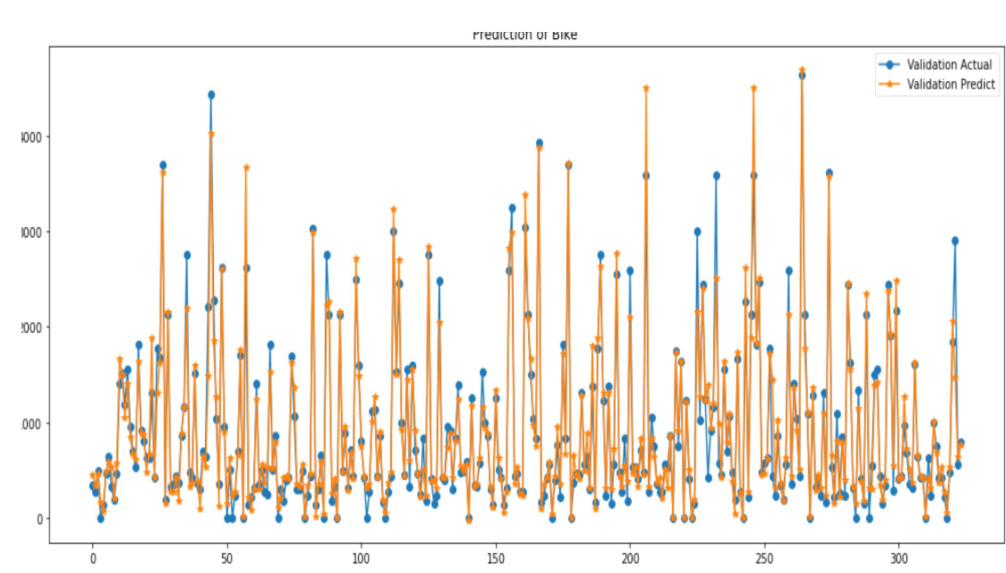
프로젝트 기획

시설물 유무 등의 라벨 생성 및 데이터 전처리

정류장 위치 추천 알고리즘 설계

모델 선정 : XGBOOST (RFE)

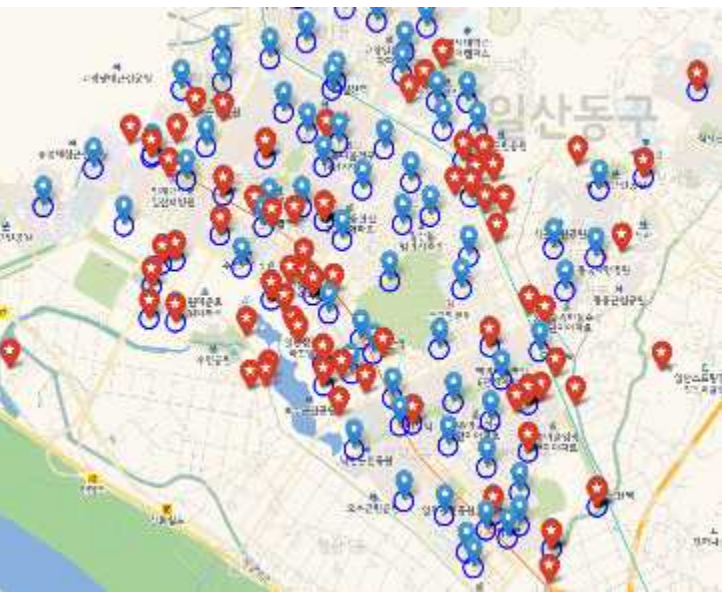
- 주거 및 유동 인구, 기타 시설물의 유무 등을 기반으로 영역 내의 예측 수요를 산출하는 모델 작성
- 100m 간격의 단위 좌표 당 예측 수요 산출
- 설명력이 가장 좋은 XGBOOST 모델 선정
- RFE로 과적합 방지 / score : 0.9379
- 유의미한 features만 남겨 적용의 용이성 도모



* 모델 예측값과 실제값의 적합도를 시각화한 자료

최적 위치 선정, 추천 정류장 산출

- 동별 적정 정류장 수를 설정해 분포가 치우치지 않게 제한
- 예측 수요가 높은 구역 내에 정류장 위치 선정
- 예측 수요를 기반으로 적정 거치대 수 산출
- 구역 내 기존 정류장 존재 시 보전, 신설 정류장 보전
- 자전거 분실, 신축 지구 정류장 신설 등의 요소 고려
- 호수 공원 부근 정류장 증설, 신축 지구 정류장 증설



* 기존 정류장과 추천 정류장을 지도에 시각화한 자료
* 파란색 마커 : 기존 정류장 / 빨간색 마커 : 추천 정류장
기존 정류장 수 : 164 개 / 추천 정류장 수 : 266 개
* 정류장 분포가 많은 호수 공원 주위를 확대한 자료

02

PROJECT. 2

감염병 유행 감지 및 개인 맞춤 정보 제공 서비스

ABOUT PROJECT

코로나
뉴노멀
데이터 크롤링
감염병 유행 감지
ARIMA
개인화 서비스 제안

링크:

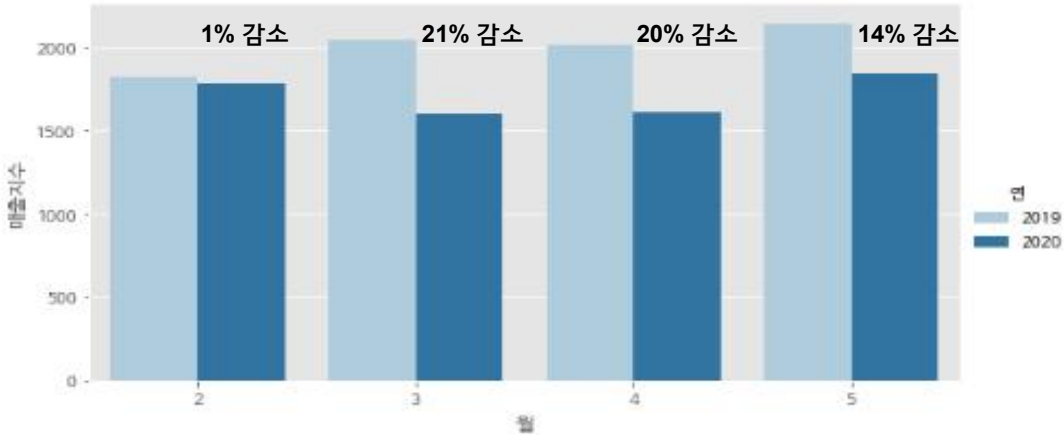
https://github.com/Neulvo/big_con

문제 정의

코로나 이후의 뉴노멀을 감염병이 일상화된 현실로 정의

감염병 정보를 개인에게 맞춤 제공함으로써 개인 안전 및 편의 도모

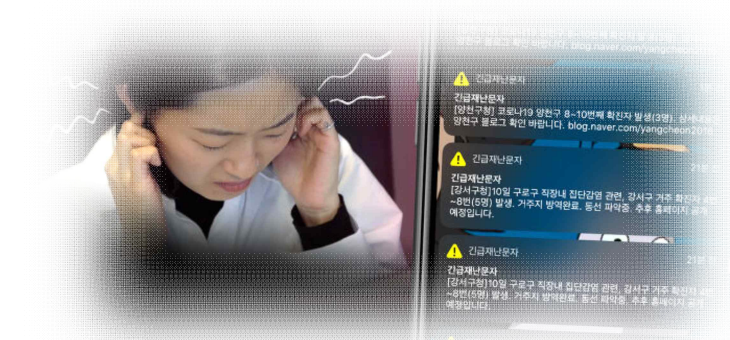
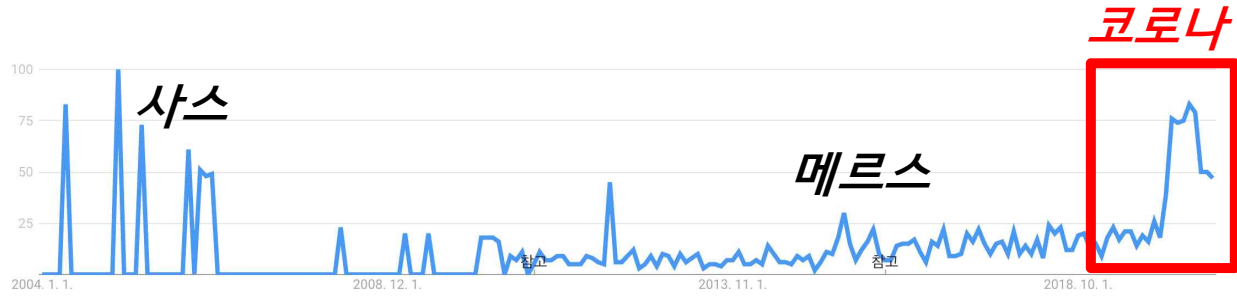
EDA 및 문제 제기



코로나 판데믹

- 코로나 위험의 확산, 인명 및 경제적 피해 발생
- 감염병 위험 지속 전망
- > 세계적 감염병의 위험성 확인, 대처 필요

* WHO CORONA-19 DashBoard 자료
2019, 2020년도 동월 대비 S카드사 매출 변화 자료

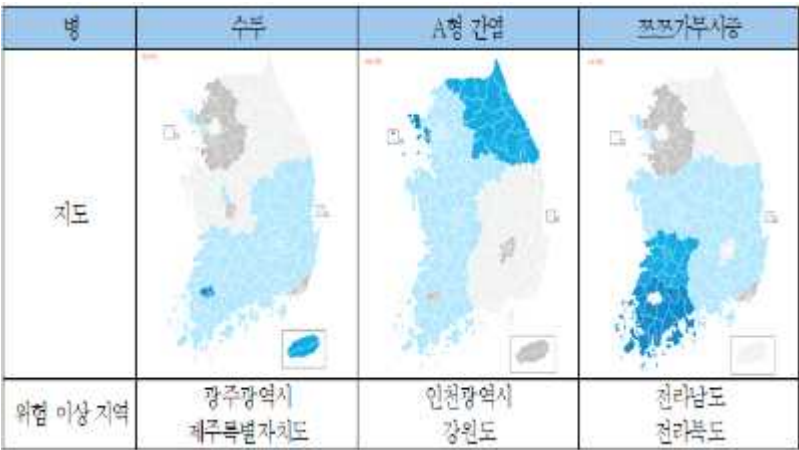


감염병 관심 증가 및 재난 문자의 문제점

- 유행 시기 감염병 관심 증가, 코로나 이후 관심 지속 전망
- 재난 문자 차단으로 보는 정보 기피 현상
- 대상 범위가 넓으면 정보에 대한 관심이 떨어질 수 있음을 발견
- > 감염병 정보의 수요 및 적절한 대상 범위 설정의 필요 확인

* Google Trend 감염병 관심도 추이 자료 (2004~2020.07)
재난 문자 차단 관련 기사 사진 자료

감염병	계절	최고비율	감염병	지역	최고비율	감염병	연령	최고비율
수두	겨울	31.2%	수두	경기	27.7%	수두	0~9세	84.4%
유행성이하선염	봄	27%	유행성이하선염	경기	23.5%	유행성이하선염	0_9세	54.8%
쯔쯔가무시증	가을	88.5%	쯔쯔가무시증	경남	16.7%	쯔쯔가무시증	70세이상	47.5%
성홍열	봄	30.7%	성홍열	경기	28.9%	성홍열	0~9세	96.6%
A형간염	봄	34%	A형간염	경기	31.1%	A형간염	30~35세	35.3%
C형간염	여름	31%	C형간염	경기	19.6%	C형간염	70세이상	42.1%
말라리아	여름	64.3%	말라리아	경기	51.2%	말라리아	20~29세	42.5%



계절, 지역, 연령 등 요인별 유행 감염병 종류 상이

- > 환경 및 개인 요인에 따른 맞춤 대응이 필요
- > 요인 기반의 개인화된 맞춤 감염병 정보 제공 서비스 착안

* 감염병 빈도수를 기반으로 각 감염병의 특징 요인을 분석한 자료
감염병 빈도수를 기반으로 유행 시기의 감염병 데이터를 시각화한 자료
2010~2020.07 국내 감염병 유행 정보를 웹크롤링하여 작성

유행 감지 모델 설계 및 개인화 서비스 제안

감염병 유행 감지

- 과거 5 개년 데이터를 기준으로 각 감염병 별로 적절한 ARIMA 모델을 찾아 fit 한 후 예측값과 신뢰구간을 산출
- 실제값이 신뢰구간을 벗어날 경우 해당 감염병의 유행 위험이 존재한다고 판단

* 하얀 원 : 감염병 발생 실측값
빨간 선 : 모델 예측값
파란 선 : 신뢰 구간 범위



* 2010~2020.07 국내 감염병 유행 정보를 기반으로 설계한 ARIMA 모델
자료는 웹크롤링으로 수집

기여 및 역할

프로젝트 기획, 아이디어 제안

감염병 유행 감지를 위한 ARIMA 모델 및 알고리즘 작성

TF-IDF 기반 개인화 서비스

- 개인 맞춤 위험 감염병 도출을 위해 단어의 특정 문서 내 중요도를 구하는 TF-IDF 방법론 활용
- 연령, 계절, 지역 요인을 고려해 TF-IDF score 산출
- Score가 높은 감염병 3종류를 도출하고 그와 관련된 안전 수칙 정보를 제공하는 알고리즘 작성

```
b = final_output(input_age = 75, input_season = '여름', location = '서울노원구')
```

-----top3-----
70세 이상, 여름, 서울노원구 조심해야할 감염병 3개~

A형간염 0.172605
장동원성대장균감염증 0.129466
유행성 이하선염 0.045933
dtype: float64
-----top3-----

※ A형간염의 예방 수칙 ※

개인위생 관리.
잘 익힌 물을 마시거나 충분한 익힌 음식을 섭취,
외출 후에는 반드시 손을 깨끗이 씻는다,
예방접종을 반드시 맞는다.

★백신을 접종하세요★

※ 장동원성대장균감염증의 예방 수칙 ※

모육장에서는 고기가 동물의 부산물에 의해 오염되지 않게 주의한다.
우유는 살균하도록 하며, 고기는 충분히 익혀서 먹는다.
간이살수도 및 수영장외의 열수 소독을 철저히 한다.
모유를 먹이는 것이 좋으며, 우유를 할 때 쓰는 물은 반드시 끓여 쓴다.
유아원, 양육원을 보육시설과 보육시설의 종사자 및 수용자에 대한 개인 위생 수칙을 준수하도록 교육한다.
과일, 야채는 깨끗한 물에 충분히 씻어서 먹는다.

※ 유행성 이하선염의 예방 수칙 ※

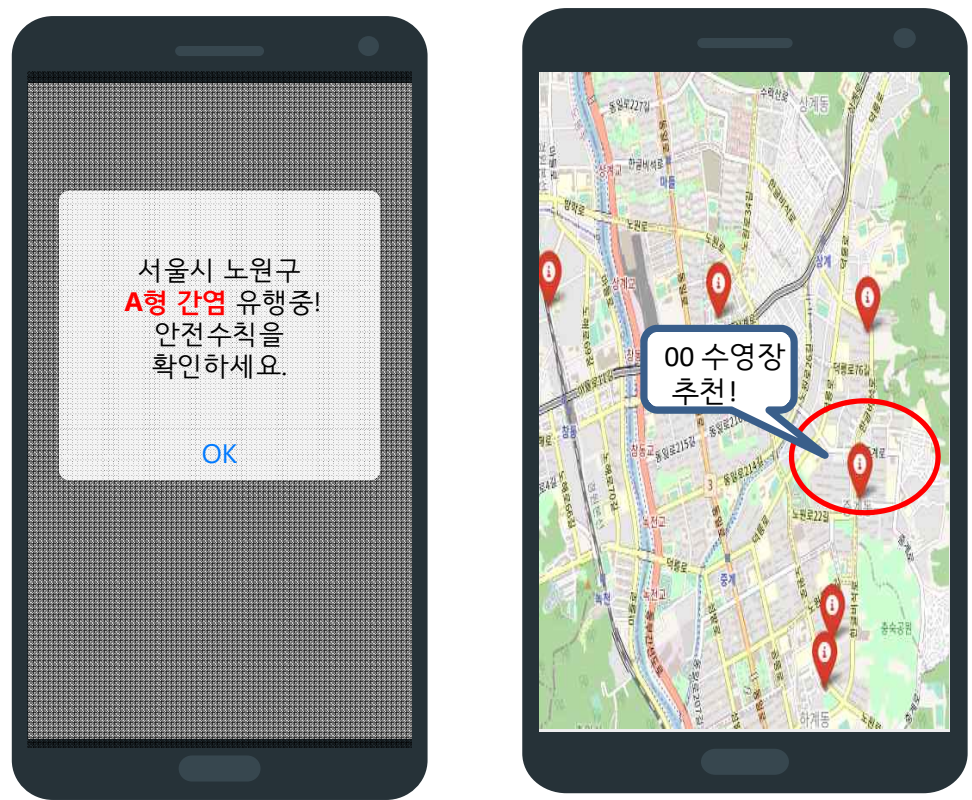
사람이 많이 모인 곳이나 밀폐된 곳을 피한다.
외출 후 곧바로 손발을 씻고 양치질을 한다.
예방접종을 반드시 맞는다.

★백신을 접종하세요★

* 2010~2020.07 국내 감염병 유행 정보를 기반으로 작성한 TF-IDF 알고리즘
자료는 웹크롤링으로 수집

서비스 제안 및 기대 효과

- 감염병 유행 감지 및 개인 맞춤 위험 감염병 정보 전달
-> 개인 위험 감소 및 유행 확산 방지 기대
- 개인 활동을 위한 안전 지역의 시설 추천
-> 개인의 안전 및 편의 도모



PROJECT. 3

논쟁적 에세이의 문장 요소 분류

03

ABOUT PROJECT

Token classification
NLP
Pytorch, Tensorflow
NER
Longformer, Ensemble

링크:

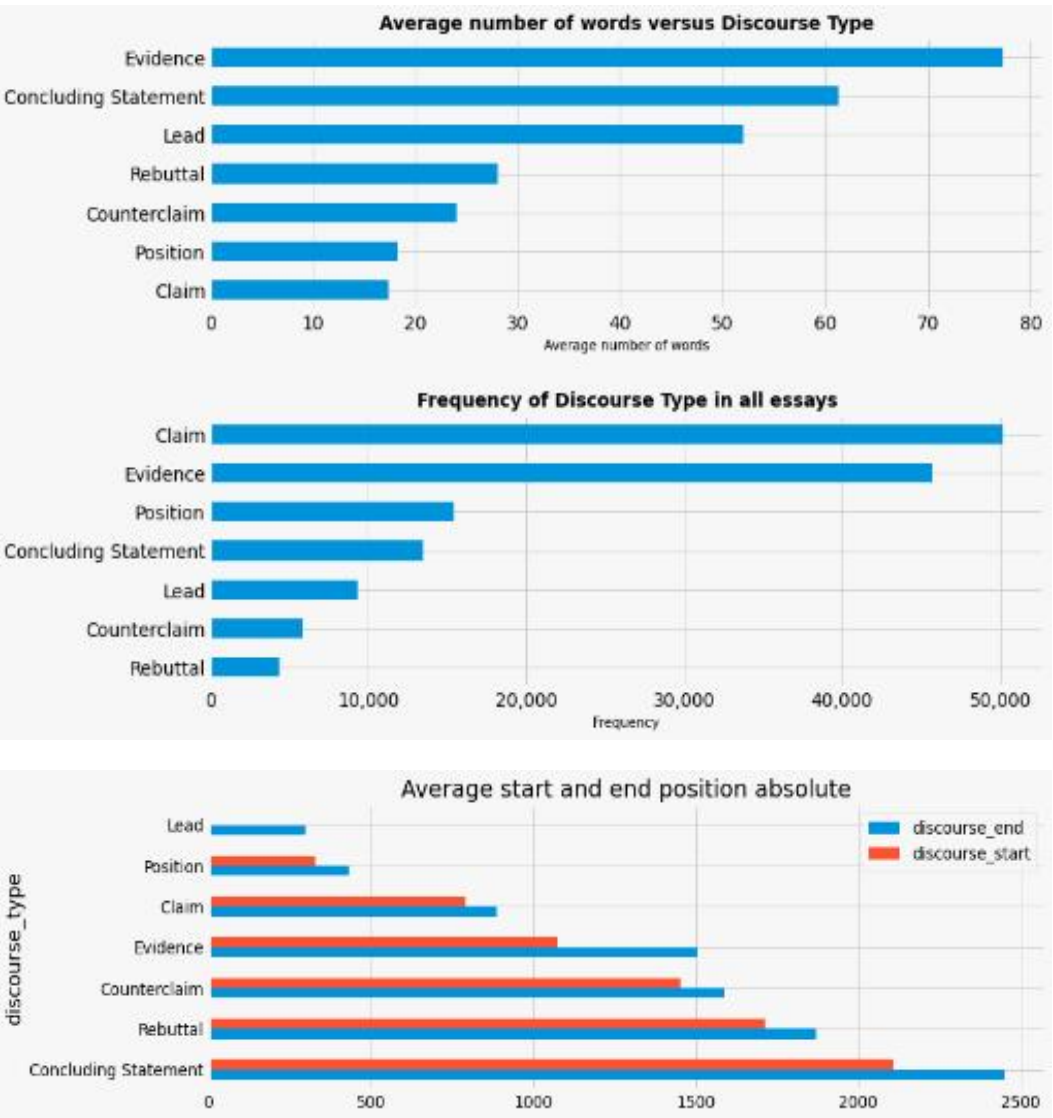
https://github.com/Neulvo/evaluate_w

문제 정의 : Token classification task

논쟁적 에세이(Argumentative essay)의 문장 요소들을 담화 유형(Discourse Type)에 따라 분류하는 문제

*데이터는 미국 6-12 grades 학생들의 에세이 데이터 (kaggle, Georgia State University 제공)

EDA 및 토큰 처리



Discourse Type EDA

- 각 Discourse Type의 단어수, 빈도수 확인
- Essay의 구성과 평균적인 Start, End position 확인

출처 : <https://youtu.be/iY2AZYdZAr0>

reports	from	Britain	and	France	that
O	O	B-LOC	O	B-LOC	O

The	European	Commission	said	on	Thursday	it	disagreed	with	German
O	B-ORG	I-ORG	O	O	O	O	O	O	B-MISC

the	Commission	's	chief	spokesman	Nikolaus	van	der	Pas	told
O	B-ORG	O	O	O	B-PER	I-PER	I-PER	I-PER	O

B-XXX labels are used at the beginning of an entity while I-XXX labels are used for the following words.

```
if "chris" not in args.save_name:
    ner_labels = {'O': 0,
                  'B-Lead': 1,
                  'I-Lead': 2,
                  'B-Position': 3,
                  'I-Position': 4,
                  'B-Evidence': 5,
                  'I-Evidence': 6,
                  'B-Claim': 7,
                  'I-Claim': 8,
                  'B-Concluding Statement': 9,
                  'I-Concluding Statement': 10,
                  'B-Counterclaim': 11,
                  'I-Counterclaim': 12,
                  'B-Rebuttal': 13,
                  'I-Rebuttal': 14}
```

토큰 태깅 : NER, Named entity recognition

- Entity가 동일한 Token들을 묶어 sequence를 분류하는 데 용이한 NER 방법론을 활용
- Discourse Type에 따라 B,I,O 라벨을 붙여 mapping (Beginning, Inside, Outside)

D72CB1C11673

Making choices in life can be very difficult. People often ask for advice when they can not decide on one thing. It's always good to ask others for their advice. **Claim** When you have multiple opinions you have the ability to make the best choice for yourself. Seeking multiple opinions can help a person make a better choice. **Position** because it can decrease stress levels, **Claim** levels, a great chance to learn something new, can be **Claim** be very helpful and beneficial. To **Claim** begin with, Seeking information from more than one person can decrease stress levels. When making a decision there is a chance that you can become very stressed and overwhelmed. Being stressed can cause a person to give up on what they are doing. Having multiple opinions from people can make choosing an option less difficult and more easier. The information that you receive from others may have a special meaning in the future. Other peoples opinion can make a person feel confident in making the right **Evidence** choice. When you ask other people for their opinion you can also learn many new **Claim** things. Everyone is different and may have more experience than you. Seeking other peoples opinion can cause you to gain a new skill or lesson. For example, someones advice could teach you how to do something the correct way. Many people are very different and have experience different things in life. Seeking advice from others can teach you a lot. We all learn from our mistakes in life, by sharing your past experiences you may prevent someone else from making the same **Evidence** mistake. Seeking others opinion can be very helpful and **Claim** beneficial. Taking other peoples advice and doing what they say may lead to a great outcome. When you receive other opinions whether they are good or bad you will be able to look at them from a different point of view. For example, When a group of people give you advice on yourself, you then know how they look at you as a person. Everyone looks at certain things from a different perspective. How someone else looks at it may be how its supposed to be viewed in life. **Evidence** In conclusion, You can learn from others experiences by seeking the advice that someone give you. Making decisions can cause stress on a person. Relating to someones advice may higher your self-esteem because you no longer feel left out. Ask others for advice when making a difficult decision. When you seek others opinion more opportunities are available fro yourself. **Concluding Statement**

Token classification 시각화

- Essay Tokens를 Discourse Type에 따라 분류하고 시각화
- 예시와 같이 새로운 Essay의 문장 요소들을 자동 분류하는 것이 목표

코드 리뷰 및 분석

- GPU 자원 이슈로 코드 리뷰 및 분석으로 방향 전환

TensorFlow Longformer BaseLine

- Tensorflow 구문의 Longformer 모델 학습 코드
 - Longformer 1024 모델 사용, 성능 한계 존재
 - GPU 자원 부족, Batch size 등의 파라미터 설정 한계
 - CounterClaime과 Rebuttal의 Score가 낮음을 확인
- 링크: <https://neulvo.tistory.com/497?category=993643>

```
In [14]: def build_model():
tokens = tf.keras.layers.Input(shape=(MAX_LEN,), name = 'tokens', dtype=tf.int32)
attention = tf.keras.layers.Input(shape=(MAX_LEN,), name = 'attention', dtype=tf.int32)

config = AutoConfig.from_pretrained(DOWNLOADED_MODEL_PATH+'config.json')
backbone = TFAutoModel.from_pretrained(DOWNLOADED_MODEL_PATH+'tf_model.h5', config=config)

h = backbone(tokens, attention_mask=attention)
h = tf.keras.layers.Dense(256, activation='relu')(x[0])
h = tf.keras.layers.Dense(15, activation='softmax', dtype='float32')(x)

model = tf.keras.Model(inputs=[tokens,attention], outputs=x)
model.compile(optimizer = tf.keras.optimizers.Adam(lr = 1e-4),
              loss = [tf.keras.losses.CategoricalCrossentropy()],
              metrics = [tf.keras.metrics.CategoricalAccuracy()])

return model

In [15]: with strategy.scope():
model = build_model()

All model checkpoint layers were used when initializing TFLongformerModel.

All the layers of TFLongformerModel were initialized from the model checkpoint at ../input/tf-lonsto
If your task is similar to the task the model of the checkpoint was trained on, you can already use
editions without further training.
/opt/conda/lib/python3.7/site-packages/keras/optimizer_v2/optimizer_v2.py:356: UserWarning: The `lr`
use `learning_rate` instead.
"The `lr` argument is deprecated, use `learning_rate` instead.")

In [26]: f1s = []
CLASSES = oof['class'].unique()
for c in CLASSES:
pred_df = oof.loc[oof['class']==c].copy()
gt_df = valid.loc[valid['discourse_type']==c].copy()
f1 = score_feedback_comp(pred_df, gt_df)
print(c,f1)
f1s.append(f1)
print()
print('Overall',np.mean(f1s))

Lead 0.8063284233496999
Position 0.6841560234725578
Claim 0.6057328286559762
Evidence 0.6816788493279887
Concluding Statement 0.7827050997782705
Counterclaim 0.4854732695970009
Rebuttal 0.39030955585464333

Overall 0.6337691528480197
```

Pytorch Longformer BaseLine

- Pytorch 구문의 Longformer 모델 학습 코드
 - folds를 나눠 서로 다른 두 개의 모델 weights를 활용
 - 학습된 모델을 불러와서 사용
 - TensorFlow baseline 코드보다 나은 성능을 보이거나 data 기반의 학습이 이루어지지 않아 성능 향상에 한계가 있었음
- 링크: <https://neulvo.tistory.com/496?category=993643>

```
In [9]: df = pd.read_csv(os.path.join("D:/Users/azvov/pyun/kagglory/feedback-prize-2021/", "sample_submission.csv"))
df_ids = df['id'].unique()

tokenizer = AutoTokenizer.from_pretrained(args1.model)
test_samples = prepare_test_data(df, tokenizer, args1)
collate = Collate(tokenizer=tokenizer)

raw_preds = []
for fold_ in range(10):
current_idx = 0
test_dataset = FeedbackDataset(test_samples, args1.max_len, tokenizer)

if fold_ < 5:
model = FeedbackModel(model_name=args1.model, num_labels=len(target_id_map) - 1)
model.load(os.path.join(args1.tez.model, f"model_{fold_}.bin"), weights_only=True)
preds_iter = model.predict(test_dataset, batch_size=args1.batch_size, collate_fn=collate)
else:
model = FeedbackModel(model_name=args2.model, num_labels=len(target_id_map) - 1)
model.load(os.path.join(args2.tez.model, f"model_{fold_+5}.bin"), weights_only=True)
preds_iter = model.predict(test_dataset, batch_size=args2.batch_size, collate_fn=collate)

current_idx += 1

for preds in preds_iter:
preds = preds.astype(np.float16)
preds = preds / 10
if fold_ == 0:
raw_preds.append(preds)
else:
raw_preds[current_idx] += preds
current_idx += 1
torch.cuda.empty_cache()
gc.collect()

100%|#####| 2/2 |00:01:00:00| 1.97it/s, stage-test|
100%|#####| 2/2 |00:00:00:00| 3.27it/s, stage-test|
100%|#####| 2/2 |00:00:00:00| 3.02it/s, stage-test|
100%|#####| 2/2 |00:00:00:00| 3.09it/s, stage-test|
100%|#####| 2/2 |00:01:00:00| 2.77it/s, stage-test|
100%|#####| 2/2 |00:00:00:00| 3.26it/s, stage-test|
100%|#####| 2/2 |00:00:00:00| 2.96it/s, stage-test|
100%|#####| 2/2 |00:00:00:00| 3.15it/s, stage-test|
100%|#####| 2/2 |00:00:00:00| 3.20it/s, stage-test|
100%|#####| 2/2 |00:00:00:00| 3.19it/s, stage-test|
```

1st & 2nd Place Solution

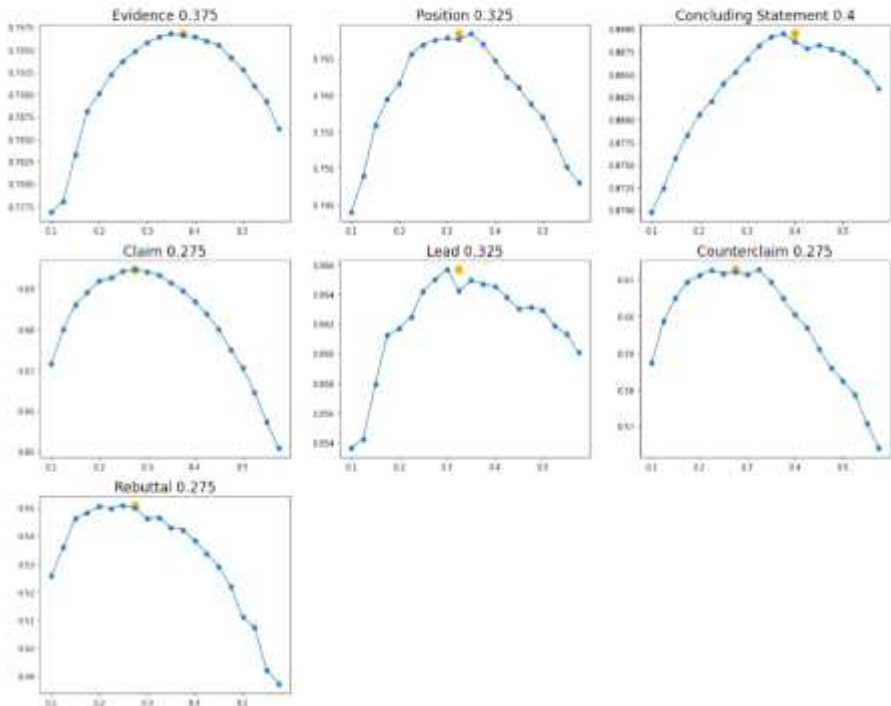
- 1st Solution : longformer base 4096 Model 사용
- 2nd Solution : WBF Ensemble Model 사용
- CV OOF score를 활용해 threshold를 설정
- Threshold 설정 및 활용이 Critical 했음
- 링크(1st): <https://neulvo.tistory.com/527?category=993643>
- 링크(2nd): <https://neulvo.tistory.com/511?category=993643>

```
test_feat['pred'] = test_pred
segment_param = {
    "Lead": {'min_proba': [0.47, 0.41], 'begin_proba': 1.00, 'min_sep': 40, 'min_length': 5},
    "Position": {'min_proba': [0.45, 0.40], 'begin_proba': 0.90, 'min_sep': 21, 'min_length': 3},
    "Evidence": {'min_proba': [0.50, 0.40], 'begin_proba': 0.55, 'min_sep': 2, 'min_length': 21},
    "Claim": {'min_proba': [0.40, 0.30], 'begin_proba': 0.90, 'min_sep': 10, 'min_length': 1},
    "Concluding Statement": {'min_proba': [0.58, 0.25], 'begin_proba': 0.93, 'min_sep': 50, 'min_length': 5},
    "Counterclaim": {'min_proba': [0.45, 0.25], 'begin_proba': 0.70, 'min_sep': 35, 'min_length': 6},
    "Rebuttal": {'min_proba': [0.37, 0.34], 'begin_proba': 0.70, 'min_sep': 45, 'min_length': 5},
}
```

Ensemble Models with WBF

We will now read in the 10 submission files generated above and apply WBF to ensemble them. After applying WBF, it is important to remove predictions with confidence score below threshold. This is explained [here](#).

If only 1 model out of 10 models makes a certain span prediction, that prediction will still be present in WBF's outputs. However that prediction will have a very low confidence score because that model's confidence score will be averaged with 9 zero confidence scores. We found optimal confidence scores per class by analyzing our CV OOF score. For each class, we vary the threshold and compute the corresponding class metric score.



What else?

Blog activity

ABOUT PROJECT

활동 기록, 코드 분석
Course 리뷰, 방법론 소개
오류 기록, 정보 소개

링크:
<https://neulvo.tistory.com/category/Data>

감사합니다 :)