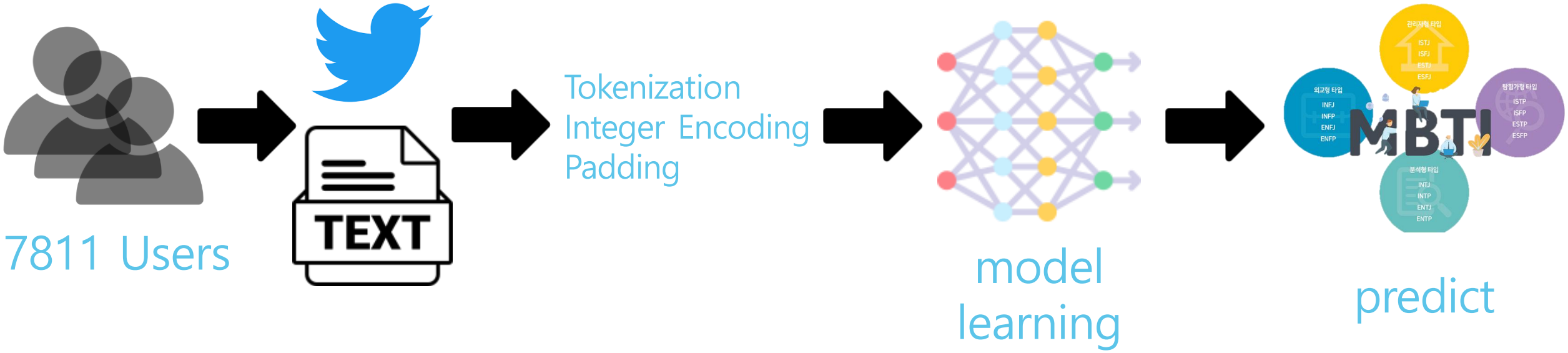
최종 보고서

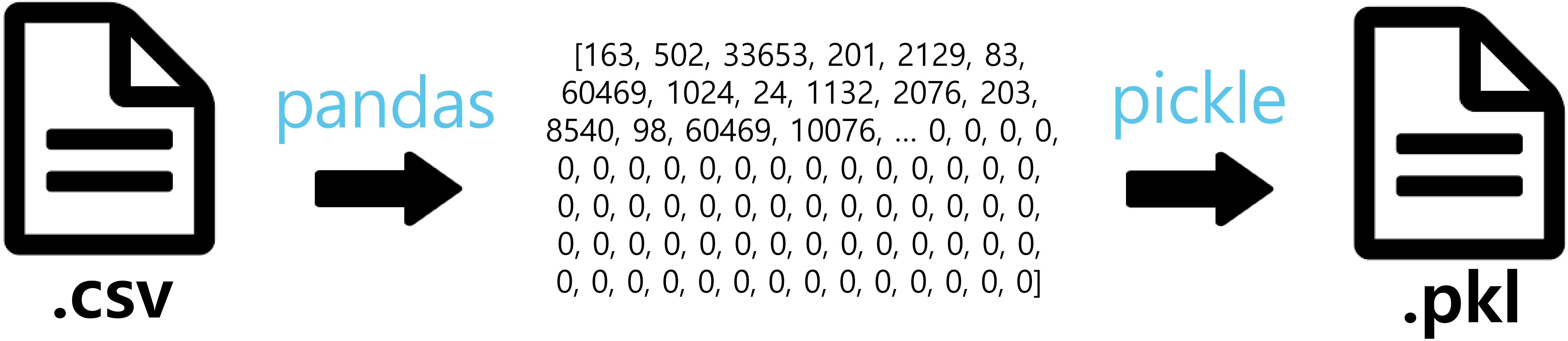
SNS의 digital footprint를 이용한 MBTI 예측

C111093 신현석
C111197 최호재
C135333 임원재
B911068 박범조

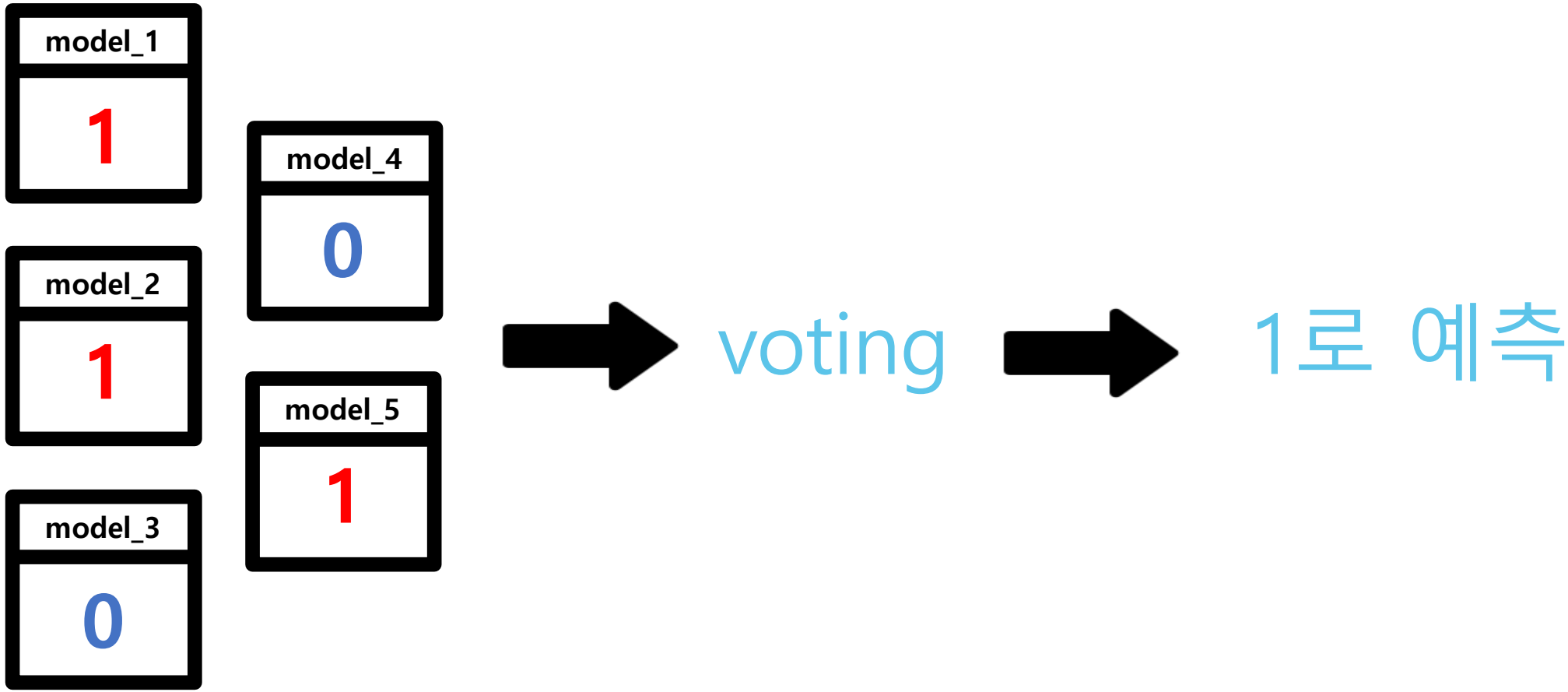
개요


7811 Users
TEXT
Tokenization
Integer Encoding
Padding
model
learning
MBTI
predict

개요


.csv
pandas
[163, 502, 33653, 201, 2129, 83, 60469, 1024, 24, 1132, 2076, 203, 8540, 98, 60469, 10076, … 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
pickle
.pkl

모델 선정 : 앙상블 기법



모델 선정 : 앙상블 기법

model 1 : TF-IDF 행렬과 코사인 유사도를 이용한 KNN

model 2 : SVM

model 3 : RNN - lstm

model 4 : XGBoost

model 5 : Random Forest

MBTI

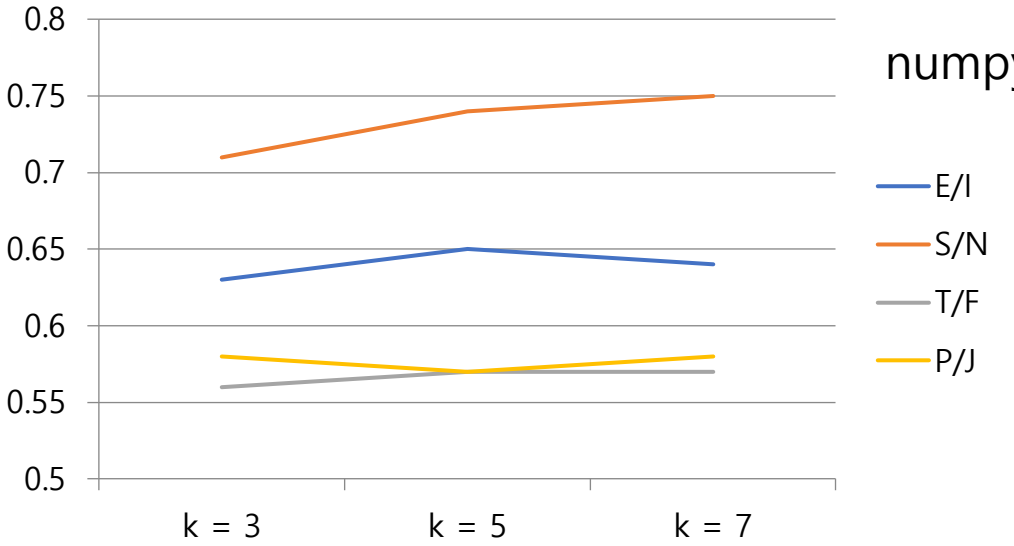
16개의 target class : {'infp', 'infj', 'intp', 'intj', 'isfp', 'isfj', 'istp', 'istj', 'enfp', 'enfj', 'entp', 'entj', 'esfp', 'esfj', 'estp', 'estj'}

4개의 target class : {'E or I', 'S or N', 'T or F', 'P or J'}

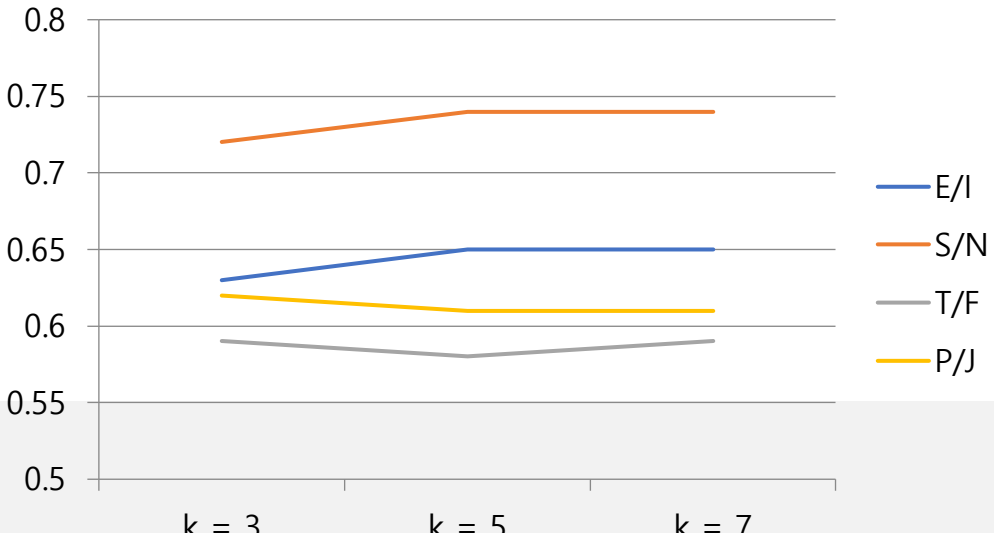
ex) istp : [1, 0, 0, 0]

model 1 결과 & 분석- TF-IDF 행렬과 코사인 유사도를 이용한 KNN

majority vote



weighted majority vote



numpy, pickle, sklearn-accuracy_score classification_report 라이브러리 사용

input_data : TF-IDF 행렬(tf 값과 idf 값을 곱한 값)

$$similarity = cos(\Theta) = \frac{A \cdot B}{||A||\,||B||}$$

코사인 유사도 사용
두 벡터 간의 코사인 각도를 이용하여 구할 수 있는 두 벡터의 유사도
각이 동일한 경우 : 1
각이 직각인 경우 : 0
각이 반대인 경우 : -1
=> 1에 가까울수록 유사도 높음

E / I : 65%
S / N : 74%
T / F : 58%
P / J : 58%

model 2 결과 & 분석- SVM

SVM : classification에 사용되는 지도학습 모델 (고차원 데이터에 효과적)

numpy, pickle, sklearn.svm 라이브러리 사용

Class 1에 민감하도록 가중치 부여

하이퍼 파라미터 grid search

param_grid = {'C': [0.1, 0.3, 0.5, 0.7, 1.0], 'class_weight': [{0: 1, 1: 2}, {0: 1, 1: 3}]}

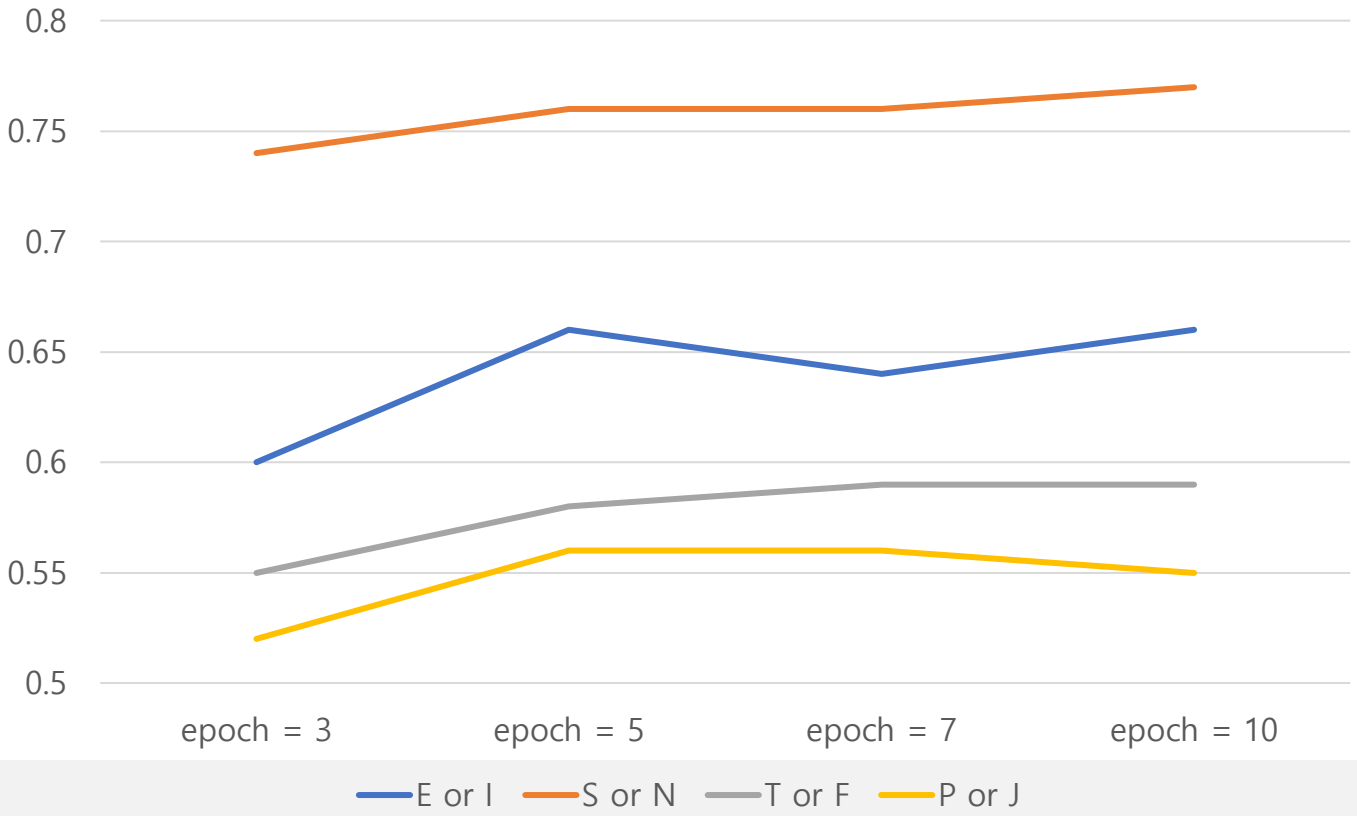
1차 predict
E / I : 67.1%
S / N : 77.7%
T / F : 59.1%
P / J : 58.2%

2차 predict
E / I : 48.2%
S / N : 54.7%
T / F : 45.1%
P / J : 57.3%

3차 predict
E / I : 59.4%
S / N : 77.5%
T / F : 44.9%
P / J : 58.2%

model 3 결과 & 분석- RNN (lstm)

LSTM : RNN의 한 종류, 자연어 처리에 사용되는 모델,
의존성을 필요로 하는 학습에 효율적
input_data : 직접 정수 인코딩 진행한 데이터



numpy, pickle, tensorflow, keras 라이브러리
사용하여 모델 import

E / I : 60%
S / N : 74%
T / F : 55%
P / J : 52%

model 4 결과 & 분석- XGBoost

numpy, pandas, sklearn, XGBoost 라이브러리 사용

learning_rate	n_estimators	max_depth	Accuracy
0.1	100	7	max

E_I 정확도: 67.4%
S_N 정확도: 78.4%
T_F 정확도: 56.7%
P_J 정확도: 56.3%

model 5 결과 & 분석- random forest

pandas, numpy, RandomForestClassifier 라이브러리 사용

n_estimators	max_depth	Accuracy
200	x	max

E_I 정확도: 68.2%
S_N 정확도: 78.4%
T_F 정확도: 61.1%
P_J 정확도: 57.9%

앙상블 - voting

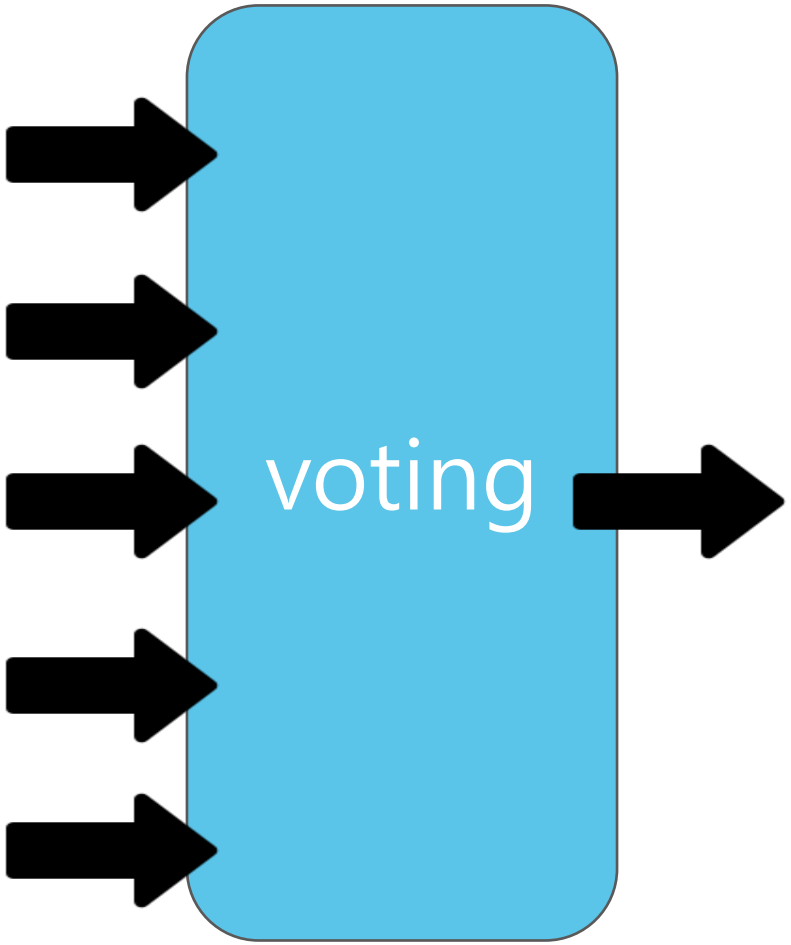
model 1 (KNN)

model 2 (SVM)

model 3 (RNN - LSTM)

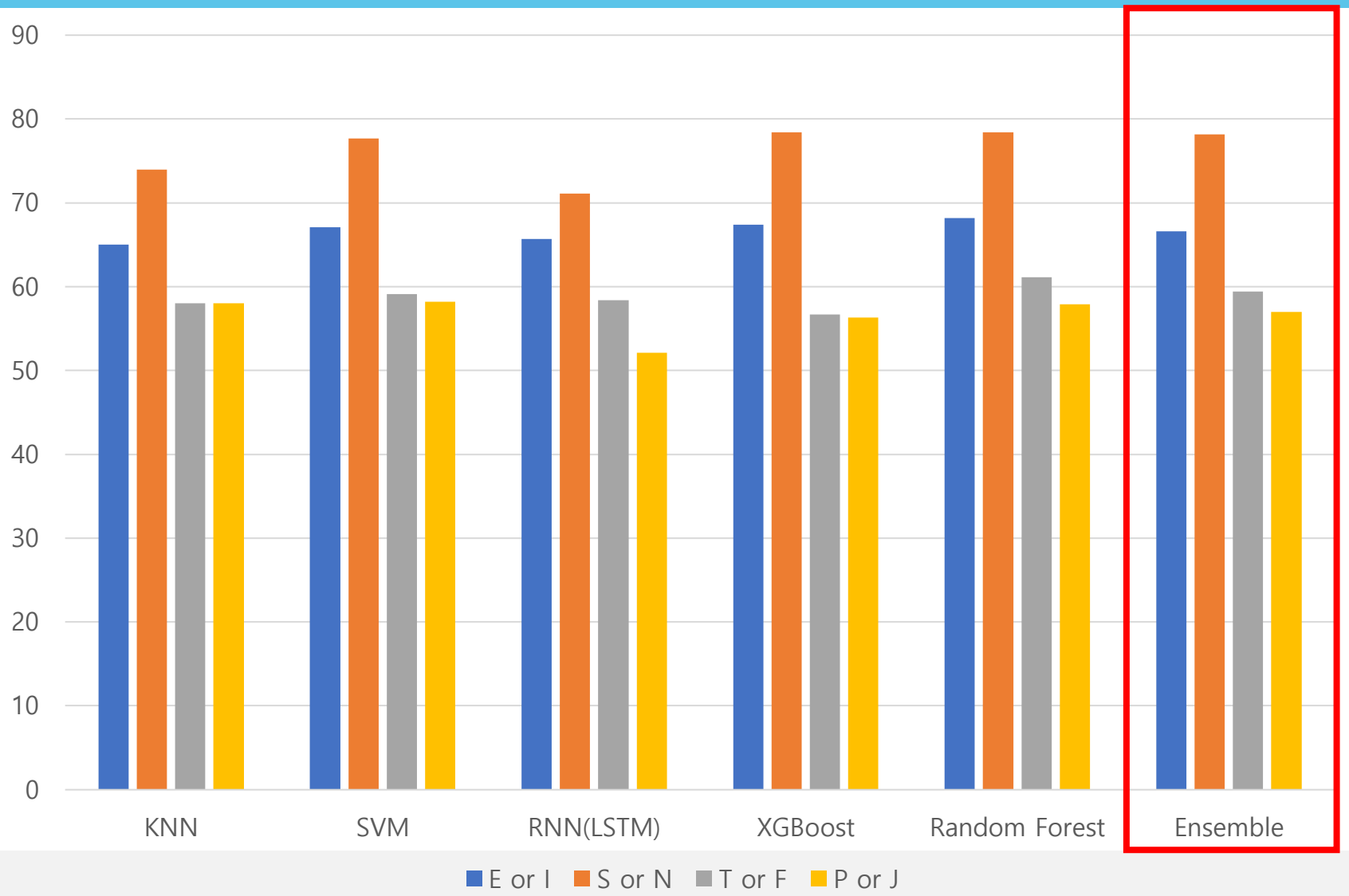
model 4 (XGBoost)

model 5 (Random Forest)



E / I : 66.6%
S / N : 78.2%
T / F : 59.4%
P / J : 57.0%

앙상블 결과 & 분석- Random Forest



Focus on

MBTI 16개의 레이블을 4개의 차원으로 mapping

- 16개의 레이블에 대한 예측값은 예상보다 낮았음
- 4개의 차원을 두고 0인지 1인지 예측하게 함
- 예측한 MBTI가 맞고 틀리다가 아닌, 어느 성향을 맞고 틀리게 판단했는지 확인 가능

N/S 레이블에서의 비교적 높은 정확도(78.2%)

- 각각 상상과 직관을 의미하는 N/S 유형은 인터넷 상에서 자신의 생각을 표현한 글은 N/S 성향을 잘 드러내어 높은 정확도를 보인다고 예상

결론

자연어를 인공지능 모델에 input할 수 있게 전처리 진행

5개의 모델을 선정하여 학습 후 예측 진행

앙상블 기법으로 5개 모델의 predict 값 중 majority 값을 최종 predict 값으로 선정

5개의 모델의 예측값의 추이가 비슷하여 기대했던 앙상블의 효과를 극적으로 내지 못함

사용자가 작성한 텍스트로 성격 유형 예측을 진행하여 유의미한 결과를 냄

E / I : 66.6%

S / N : 78.2%

T / F : 59.4%

P / J : 57.0%

기여한 점

C111093 신현석 : 정수 인코딩 구현, KNN 모델, 자료 생성

C111197 최호재 : 발표 진행, SVM 모델, 자료 생성

C135333 임원재 : 정수 인코딩 구현, RNN(LSTM) 모델, 앙상블 진행

B911068 박범조 : 주제 선정, XGBoost, Random Forest 모델, 자료 생성