Where is this hamburger from?

CLASSIFYING REDDIT POSTS BETWEEN MCDONALD'S & BURGER KING

Kemalcan Alaeddinoglu
5/28/2020

Courtesy to David Buenos Aires and Burger King

Designing classification models to identify the correct class of the posts between McDonald's and Burger King subreddits

AGENDA

1. Data Acquisition and Data Cleaning
2. Modeling
3. Scores
4. Best Parameters
5. Conclusion

1. Data Acquisition and Data Cleaning

Pushshift API

500 comments x 12 times = 6,000 comments

subreddit	
BurgerKing	3476
McDonalds	1876

Kept train dataset 67% when splitting.

2. Modeling

Transformer

TF-IDF vectorizer

Estimators

Multinomial Naive Bayes, K-Nearest Neighbor, Logistic Regression

Hyper-parameters

TF-IDF stop_words: [“none” ,“english”] ngram_range: [(1,1),(1,2),(1,3)]

MNB alpha: [0.01, 0.1, 1]

KNN n_neighbors: [3,5,7] p: [1,2]

LR penalty : [“𝓁1”,“𝓁 2”] solver:[“liblinear”] C:[5.5, 6]

2. Modeling

```
# Instantiating Pipeline with transformer and estimator
pipe = Pipeline([
    ("tfdif",TfidfVectorizer()), #Tfidf will be our transformer
    ("mnb", MultinomialNB())
    # multinomial NB model will be used as estimator
                ])
```

```
# parameters dictionary
pipe_params = {
    'tfdif__stop_words': ["none", 'english'],
    'tfdif__ngram_range':[(1,1),(1,2), (1,3)],
    "mnb__alpha":[0.01, 0.1, 1]
}
```

```
# instantiating Gridsearch with pipe, and given parameters. Shows limited parameters (verbose=1)
gs = GridSearchCV(pipe, param_grid=pipe_params, verbose=1, cv=5 )
```

3. Scores

Transformer	Estimator	Best Score	Testing Data Score	ROC_AUC Score	Sensitivity Score	Specificity Score
TF-IDF Vectorizer	Multinomial Naive Bayes	98%	99%	99%	98%	100%
TF-IDF Vectorizer	K-Nearest Neighbors	97%	99%	99.5%	99%	99.5%
TF-IDF Vectorizer	Logistic Regression	99%	97%	99%	98%	99.5%

The model's baseline accuracy is 35%

3. Scores

Transformer	Estimator	Best Score	Testing Data Score	ROC_AUC Score	Sensitivity Score	Specificity Score
TF-IDF Vectorizer	Multinomial Naive Bayes	98%	99%	99%	98%	100%
TF-IDF Vectorizer	K-Nearest Neighbors	97%	99%	99.5%	99%	99.5%
TF-IDF Vectorizer	Logistic Regression	99%	97%	99%	98%	99.5%
TF-IDF Vectorizer	Multinomial Naive Bayes*	99%	99%	99%	100%	99%
TF-IDF Vectorizer	K-Nearest Neighbors*	98%	99%	99%	99%	100%
TF-IDF Vectorizer	Logistic Regression*	99%	99%	100%	99%	99%

*stop_words = ["McDonalds", "McDonald's", "whopper", "Big Mac", "King", "loving", "BK"]

4. Best Parameters

TF-IDF & Multinomial Naive Bayes,

“ngram_range” : (1, 2) and “English” stopwords for the transformer

“alpha” : 0.01 for the estimator

TF-IDF & K-Nearest Neighbor

“ngram_range” : (1, 1) and “English” stopwords for the transformer

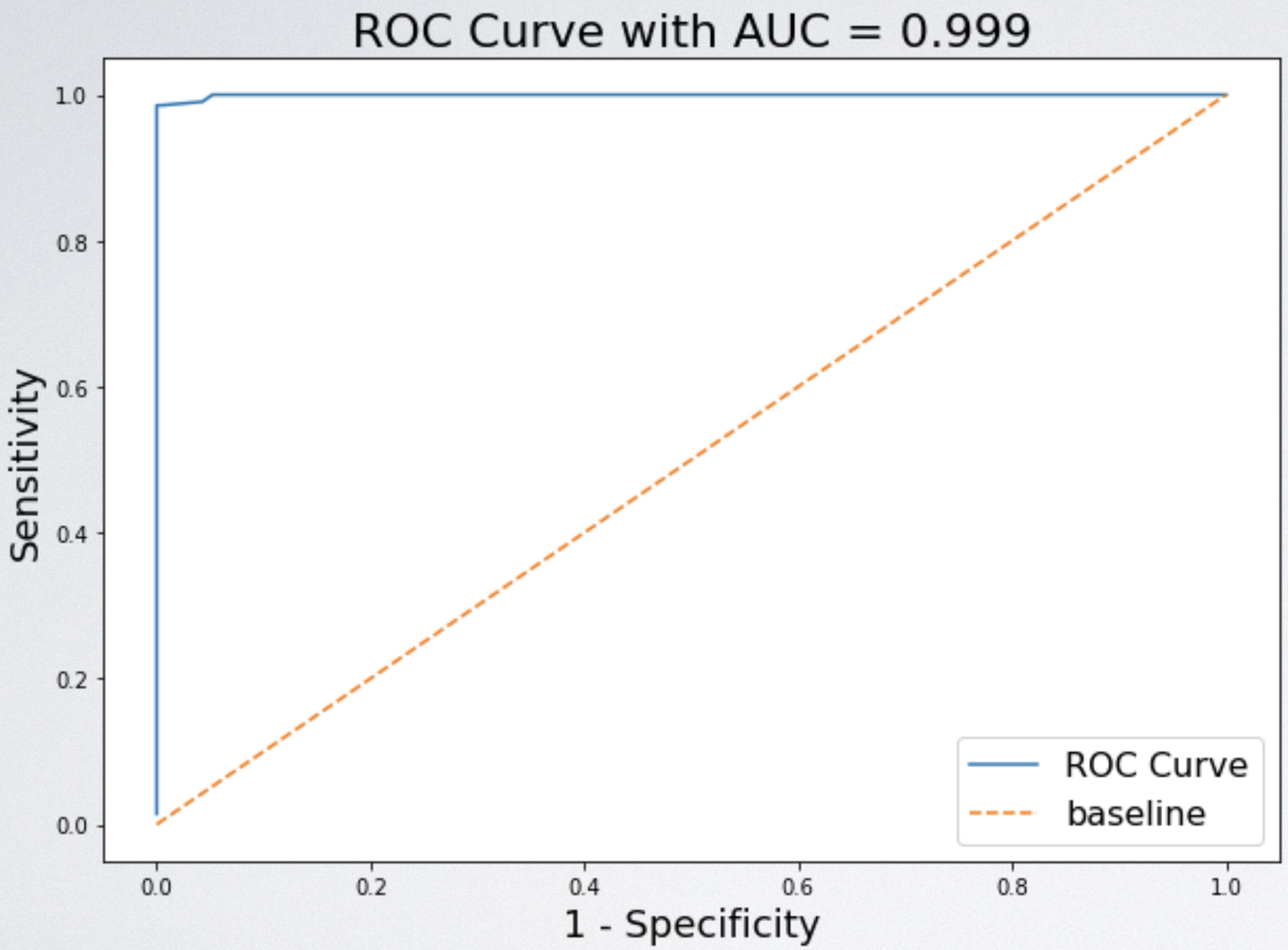
“n_neighbors” : 3 and p: 1 for the estimator

TF-IDF & Logistic Regression

“ngram_range” : (1, 1) and “English” stopwords for the transformer

“penalty” : “l1”, solver: “liblinear” and C:5.5 for the estimator

4. Conclusion



- All 3 models are very close to the perfect accuracy.
- Using manually created stop words did not change the score
- Changing the balance did not change the score