영문 기사 Classification을 통한 모델의 정확도 비교 및 분석

Comparison and Analysis of Model Accuracy through Classification of English Articles

2017313260 수학과 이재민

요 약

최근에는 컴퓨터 비전, 음성, 생체 인식 등 다양한 컴퓨터 과학 분야에서 classifier 연구가 활발히 이루어지고 있고, 다양한 타입의 모델이 개발되어지고 있다. 그런데 모델의 성능은 분류할 데이터의 특성이나 모델의 매개변수, 환경에 따라 크게 달라진다. 단일 모델로 주어진 모든 문제에서 고성능을 요구하는 데에는 한계가 있다. 본 연구에서는 영문 기사의 주제를 분류하는 상황에서 각종 classifier 모델을 적용한 뒤 성능을 비교하고, 그에 따라 각 모델을 분석한다. 본 연구의 실험 결과는 단순한 classifier가 강점을 보이고, NB classifier가 가장 좋은 성능을 보였다.

1. 서 론

Classification 모델은 supervised learning에 기반하여 기존에 존재하는 데이터와 그에 해당하는 라벨의 관계를 학습해서 새로운 데이터에 대한 라벨을 찾아주는 모델이다. 최근에는 컴퓨터 비전, 음성, 생체 인식 등 다양한 컴퓨터 과학 분야에서 연구가 활발히 이루어지고 있고, 다양한 타입의 모델이 개발되어지고 있다. 그런데 모델의 성능은 분류할 데이터의 특성이나 모델의 매개변수, 환경에 따라 크게 달라진다. 단일 모델로 주어진 모든 문제에서 고성능을 요구하는 데에는 한계가 있다.

본 연구에서는 영문 기사의 라벨, 즉 주제를 분류하는 것으로, 유명한 classification 모델의 성능을 분석할 것이다. 세 가지의 모델을 이용했는데, 이들은 K-NN, Multinomial Naïve Bayes classifier, Multi-layer perceptron(MLP) 모델이다. 또한 보다 정확한 성능 비교를 위해 Accuracy만이 아닌 f1-score를 활용하기로 했다.

2. 본 론

2.1. 관련 연구

특정한 상황에 대해 classifier의 성능을 비교하는 연구들이 다수 존재한다. 그 중에서도 학생들의 성적을 조기에 예측해서 성과를 개선하기 위해 어떤 교육 데이터 세트에 대한 학생 성과 예측모델에 대해서 연구한 논문[1]이 있다. 이는 본 연구와 비슷하게 KNN과 Naïve Bayes classifier 모델의 성능을 비교했고, Naïve Bayes가 더 낫다는 결론을 보여주었다. 본 연구에서는 영문 기사 classification이라는 특수한 상황과, MLP 모델을 추가로 도입해서 비교했다.

2.2 이론

본 연구는 우선 영문 기사를 단어로 나누어 TF값과 TF-IDF값을 입력해서 모델을 이용했다.

TF: 단어가 나타난 횟수를 의미한다

TF-IDF: 단어의 빈도에 비례하고 문서의 빈도에 반비례하는 값이다. 이는 여러 문서로 이루어진 문서군이 있을 때 어떤 단어가 특정 문서 내에서 얼마나 중요한 것인지를 나타내는 통계적 수치이다.

다음은 이용한 모델들에 대한 이론이다.

K-NN[2]: k-최근접 이웃 알고리즘 (k-nearest neighbor)알고리즘을 이용한 모델이다. 이는 k값 외의 특별한 매개변수를 주지 않아도 되며, 오직 데이터끼리 분석을 해서 예측하는 모델이다. 단, 데이터간 거리를 구하는 방식은 선택할 수 있다. 그래서 모든 데이터간 연산이 끝날 때까지 지연되므로 lazy learning의 일종이다. 이는 가장 간단한 기계학습 알고리즘이라고 볼 수 있다. 거리에 의존하기 때문에 트레이닝 데이터를 정규화하면 정확도를 크게 향상시킬 수 있다.[3] 데이터의 지역구조에 민감하다는 단점이 있다.

NB classifier[4]: Multinomial Naïve Bayes classifier을 이용했다. 이는 특성들 사이의 독립을 가정하는 Bayes 정리를 이용한 확률 classifier의 일종이다. 이는 기존의 트레이닝 데이터의 양이 적어도 효율적인 작동이 가능하다. 또한 간단한 디자인으로도 복잡한 상황에서 잘 작동한다.

MLP classifier[5]: 다층 퍼셉트론(multi-layer perceptron)을 이용한 classifier 모델이다. input 레이어, hidden 레이어 및 output 레이어의 3개 이상의 노드 레이어로 구성된다. 입력 노드를 제외하고 각 노드는 비선형 활성화 함수를 사용하는 뉴런이다. 이는 back-propagation을 이용한다[6]. 선형으로 분리할 수 없는 데이터를 구별할 수 있다. [7]

2.3. 성능 비교를 위한 값

Macro averaging 방식의 경우 각 원소에 대해서 구하고자 하는 측도를 구한 후, 구한 측도들의 평균을 내어 구한다

Accuracy: 모델의 직접적인 정확성을 의미한다. 이는 (옳게 예측한 데이터 수/총 데이터 수)로 구할 수 있다.
Macro averaging F1-score: F1-score는 Precision과 Recall의 조화평균을 의미한다.
이 두 수치를 사용할 것이다. Precision과 Recall의 의미는
Precision: 모델이 True라고 분류한 것 중에서 실제 True인 것의 비율이다.
Recall: 실제 True인 것 중에서 모델이 True로 예측한 것의 비율을 의미한다.

3. 실험

3.1 실험 환경

본 연구는 BBC의 영문기사 1490개를 이용했으며, business, politics, sport, tech, entertainment의 label을 가진 기사가 섞여져 있다. 주어진 데이터를 랜덤으로 80%는 트레이닝 데이터로, 20%는 테스트 데이터로 이용했다. K-NN은 TF값을 이용했고, NB모델과 MLP모델은 TF-IDF값을 이용해서 classification시켰다. 또한 각 모델은 직접 구현한 모델을 이용하였고, K-NN과 NB모델은 보다 정확한 비교를 위해 구현상의 문제를 해결한 sklearn을 import해서 확인해보았다.
K-NN: 이웃의 개수, 즉 k값을 1~17까지 소수를 선택해서 모델을 작동시켰다. 또한 직접 구현한 모델에서는 cosine similarity을 이용해 이웃을 판단할 것인지, 혹은 Manhattan Distance를 이용해 이웃을 판단할 것인지 선택할 수 있다.
NB classifier: smoothing 하는 과정에서 alpha값을 1로 고정하고 실험을 진행했다. (Laplacian smoothing)
MLP classifier: learning rate와 learning 횟수, 그리고 hidden 레이어의 횟수를 조정하면서 실험을 진행했다.

3.2. 직접 만든 모델의 실험 결과

실험 결과에서 F1-score은 Macro Averaging F1-score로 통일한다.
<표 1>은 직접 만든 K-NN 모델을 이용한 결과이다. 여기에서 m은 Manhattan distance를 이용한 결과, c 는 cosine similarity를 이용한 결과이다. Manhattan distance를 이용했을 때, 정확도가 50~69%정도로 나쁘지 않게 나왔으나, 너무 단순한 거리 방식으로 정확도가 낮게 나왔다. Cosine similarity를 이용한 결과에서는 전체적으로 정확도가 낮게 나왔다. 이는 cosine similarity가 문서의 유사성을 비교하기엔 적절하나 라벨을 분류하기에는 적합하지 않음을 보여준다.

metric \ k	1	3	5	7	11	13	17
m	**69.13**	59.73	56.71	53.02	64.43	55.70	62.08
c	**43.96**	38.64	36.25	28.86	23.49	25.50	23.83

<표 1> K-NN

직접 만든 NB classifier 모델은 정확도, precision, recall, F1-score 모두가 40%가 나왔다. 알고리즘에서의 잘못된 점은 없다고 생각하나, 좀 더 정확한 비교를 위해 sklearn을 import해서 비교해 보기로 했다.
<표 2>는 직접 만든 MLP 모델이다. 이 모델은 준수한 정확도를 보이고 있다. Hidden layer 수가 늘어날수록 정확도가 증가했다. 또한 learning 횟수가 증가했다. 150개의 샘플을 뽑아 진행했을 때, learning rate가 줄어들고, learning 횟수가 늘어나면 정확도가 95% 이상까지 나왔다. Overfitting 되지 않는 선에서 변수를 조정하면 정확도를 더 올릴 수 있을 것이다. 그러나 이 모델은 최소 3시간 이상의 학습 시간을 보여, 많은 시간이 소모된다.

변수 \ 성능		Accuracy	F1-score
Hidden layer	5000	86.21	87.33
Learning rate	0.002		
Learnin 횟수	10000		
Hidden layer	15000	89.65	90.14
Learning rate	0.002		
Learnin 횟수	10000		
Hidden layer	15000	**93.10**	**93.30**
Learning rate	0.002		
Learnin 횟수	20000		

<표 2> NB classifier

3.3. sklearn에서 import한 모델의 실험 결과

<표 3>은 sklearn에서 import한 K-NN 모델을 이용한 결과이다. 보다 적당하고 복잡한 거리를 도입했기 때문에 k=1일때는 96.67%의 정확도로, 매우 준수하다.

성능 \ k	1	3	7
Accuracy	**96.67**	93.03	91.67
F1-score	**96.62**	92.99	91.66

<표 3> K-NN; sklearn

sklearn에서 import한 NB classifier모델의 정확도는 무려 97.98%가 나왔다.

3.4. 종합

NB classifier 모델의 정확도가 가장 높고, 실행시간도 짧다. MLP모델은 주어진 데이터양이 엄청나게 많지 않고, 단순하기 때문에 복잡한 이 모델은 좋은 성능을 끌어내기 힘들어서 오히려 성능이 감소되는 모습을 보였다.

4. 결 론

본 연구에서는 영문 기사의 주제를 하기 위해 유명한 세 가지의 K-NN, Multinomial Naïve Bayes classifier, Multi-layer perceptron(MLP) 모델을 이용했다. 기존의 연구[1]과 같이 NB classifier의 성능이 가장 좋게 나왔다. 그리고 예상과 달리, 보다 복잡한 거리를 부여했을 때 K-NN의 성능이 크게 떨어지지는 않았고, MLP가 성능이 낮게 나왔다. 이러한 것으로 보아 영문 기사 classification과 같이 트레이닝 데이터가 그렇게 많지 않고, 단순한 구조를 갖고 있을 때에는 복잡한 모델보다는 오히려 단순한 모델의 성능이 좋음을 알 수 있다.

5. 참고 문헌

[1] Ihsan A. Abu Amra, Ashraf Y. A. Maghari. “Students performance prediction using KNN and Naïve Bayesian” IEEE (2017)
[2] Altman, N. S. “An introduction to kernel and nearest-neighbor nonparametric regression”. The American Statistician (1992)
[3] Piryonesi S. Madeh, El-Diraby Tamer E. "Role of Data Analytics in Infrastructure Asset Management: Overcoming Data Size and Quality Problems". Journal of Transportation Engineering, Part B: Pavements. (2020)
[4] Frank, Eibe, Remco R. Bouckaert. "Naive bayes for text classification with unbalanced classes." European Conference on Principles of Data Mining and Knowledge Discovery. Springer, Berlin, Heidelberg (2006)
[5] Hastie, Trevor. Tibshirani, Robert. Friedman, Jerome. “The Elements of Statistical Learning: Data Mining, Inference, and Prediction”. Springer, New York, NY, (2009)
[6] Rosenblatt, Frank. x. “Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms.” Spartan Books, Washington DC, (1961)
[7] Cybenko, G. “Approximation by superpositions of a sigmoidal function Mathematics of Control, Signals, and Systems” (1989)