2018008304 컴퓨터소프트웨어학부 박경하

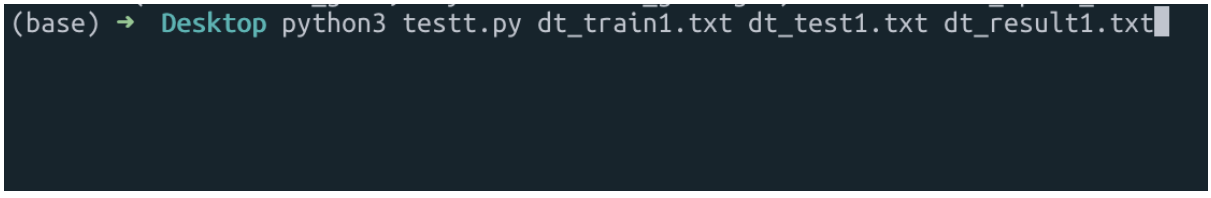
Summary of your algorithm

인풋 받은 전체 데이터를 다중 딕셔너리의 형태로 하여 decision tree 를 만들었습니다. Decision tree 를 만드는 과정에서, 각각의 feature 에 대해 information gain 을 계산해 가장 큰 값을 가지는 feature 로 데이터를 분기해나갔습니다. 트레이닝 데이터로 만든 tree 로, 테스트 데이터를 대조하여 label 을 찾아나갑니다. 그 과정에서 트리에 저장되어있지 않은 경우 현재노드의 class label 을 비교한 뒤 가장 많은 수를 갖고 있는 label 로 classification 하였습니다.

Detailed description of your codes (for each function)

load_training_data() : 트레이닝 데이터를 pandas 로 할당하는 함수입니다.
calculate_entropy(target_colums) : argument 에 대한 엔트로피를 구하는 함수 입니다
select_split_feature(feature) : info_gain 을 구해 가장 적합한 feature 을 선택하는 함수입니다. 전체 엔트로피를 구하고, 각각 feature 에 대한 가중 엔트로피를 구한 뒤, 가장 높은 info_gain 의 값을 가지는 feautre 을 리턴합니다.
build_tree(data, feature_name, parent_label) : decision tree 를 구축하는 함수입니다. 데이터가 없는 경우, 원래 데이터에서의 label 값을 다수결로 결정하며, 더 이상 분기할 feature 가 없는 경우를 종료조건으로 두었습니다. 분기할 수 있는 경우에는 가장 적합한 feature 을 고른 뒤 회귀함수를 이용하여 딕셔너리의 형태로 트리를 만들었습니다.
get_label_average(feature, inputs) : 테스트 데이터를 classify 할 때, 디시전 트리의 가지에 feature 값이 저장되어있지 않은 경우 부모 노드의 class label 의 값을 다수결로 결정하는 함수입니다
load_test_data() : 테스트 데이터를 pandas 로 할당하는 함수입니다.
classify_label(dt, inputs) : 테스트 데이터의 label 을 결정하는 함수입니다. Dict 타입이 아닌 경우를 leaf 노드라고 판단하여 그 때의 label 값을 리턴하는 함수 입니다.
write_txtfile() : 주어진 형식에 따라 class label 을 결정하고 텍스트 파일로 만드는 함수 입니다.

Instructions for compiling your source codes at TA's computer



Cmd 창에서 위와 같은 명령어로 컴파일 하였습니다.