Assignment 2

Team Member: Jing Liu, 18231917
Miao Li, 18230232
Course: Data Analytics

Introduction

This assignment majorly using majority vote and David & Skene function to produce the label of each sentence and using decision tree to training the data and produce the prediction.

Characteristics of two samples

For the first sample gold_sample, here I use random.sample to random extract 1000 sentence as sample to training, and Extract rows from gold_sample that correspond to the sentence ids and get the rows, there are totally 1000 rows for gold_sample and the distribution of labels are positive label are : and negative label are:. For the select way I use random.sample to randomly select the 1000 rows from the dataset , 486 rows are positive and 515 rows are negative data, using random.sample can keep the pos and neg are 1:1 as much as possible and in voiding of there are most of sentence are negative and only few sentence are positive. For mturk_sample, there are totally 5552 rows data and the positive are 2953 and the negative are 2599, the relationship between two datasets are finding all ids which same to gold_sample and keep all rows, so each sentence in mturk dataset has multiple labels,but the gold_sample only have one label for each sample. As the graph show s below there is little difference between the number of positive and negative graphs.

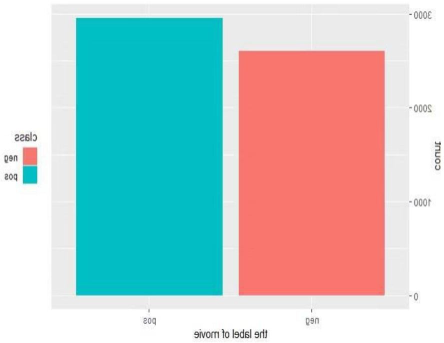

Figure 1.1 bar graph of label mturk_dataset

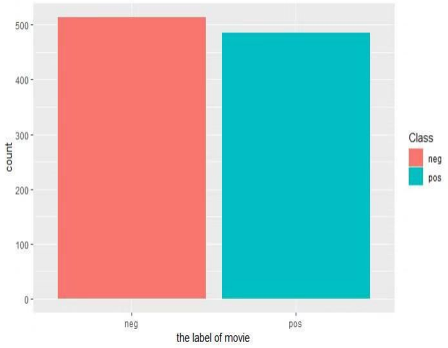

Figure 1.2 bar graph of labels gold_sample

	gold_sample dataset	mturk_sample. dataset
total number of labels	Total:1000	Total:5552
distribution of labels	Pos: 486 Neg:514	Pos: 2953 Neg:2599
average number of workers per sentence	average :1	average :5.552

Characteristics of three models

(1) ***Decision Tree with original dataset***
The decision tree is a decision analysis method that evaluates the project risk and judges its feasibility by constructing a decision tree to obtain the probability that the expected value of the net present value is greater than or equal to zero, based on the known probability of occurrence of various situations. [1] Here using the sample data as training data, the decision tree will select the best attribution to as one of node through calculating information gain, then using the fixed attribute probability to calculating the final label of the sample. Here firstly using classdecisiontreeclassifier to get the classifier and training the data by fit() function and then get the predictions, The performance of decision tree with original data: the accuracy is 52.7634487841, the F1 score are 0.503485670023, which means there are 52.7 are correctly classified.

(2) ***Decision Tree with majority vote***
The method of majority vote is a way that get the best label for a sentence through comparing the votes of each worker, if votes of positive greater than votes of negative then the real label of this sentence will be positive, in this way the label can determined by majority votes, then update the mturk_dataset with 1000 rows and the labels update to the new label after majority vote, then put the training data to the decision tree model and get the prediction, the performance for mturk _dataset are the accuracy is 53.5286530311 and the F1_score is 0.540619307832.

Decision Tree with David & Skene
The David&Skene method is another way to get the new label from crowdsourced data, firstly get two matrix from the mturk_dataset, the first one is Input Data Matrix for all workers vote for all movies, another one is the initial majority dataset for each movie, then through calculating each data in and initial input and Initialize with Majority Vote dataset by the way on slides to update the data for max iterations or convergence, so the max probability will be bigger while the smaller probability will be smaller and lastly each sentence will get a probability < 0.5 and another one > 0.5, using the probability > 0.5 to get the final label. Then using the dataset after David & Skene as training date to train the mode of decision tree. The performance of this model are: the accuracy is 50.5988575640317 and the F1-score is : 0.5360789063851877;

(3)

Comparison with tables and confusion matrices

As the result shows in the table, we ca see that the accuracy of decision tree is the lowest, which is only (52.7634487841). The accuracy is improved based on the majority vote. However, the accuracy of David&Skene is 50.5%, this is because the theory of this model is to increase the weight of another who has higher reliability and decrease the weight of another who has low reliability. Therefore the label of movie is mostly depended on reliable another. This is the reason. The result is as figure 3.1: It is clear that TP is 1197, TN is 1549, which means this model make the number of right prediction is 1197+1549 = 2746; The FP is 1512 and NP is 1169, which means the model makes wrong prediction is 1512 + 1169 = 2681; F1-score = 2*(precision * recall)/precision + recall = 0.50;

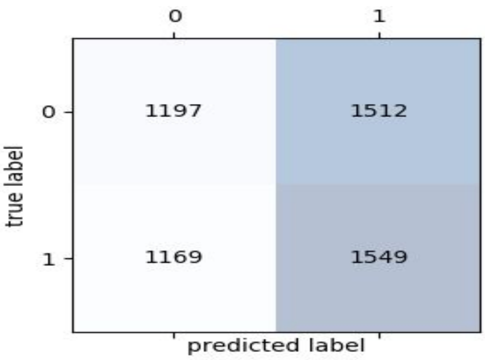

Figure 3.1 Confusion matrix of David&Skene

The confusion matrix of majority vote model is as figure 3.2: It is clear that TP is 1421, TN is 1484, which means this model make the number of right prediction is 1421+1484 = 2905; The FP is 1512 and NP is 1169, which means the model makes wrong prediction is 1288 + 1234 = 2522; F1-score = 2*(precision * recall)/precision + recall = 0.54; This model is higher than David&Skene; I believe the reason why majority vote is higher than David&Skene is because the way I process majority vote. As I commend in code that if a movie has three positive vote and three negative vote then I select the first value and drop the second. This might be the reason why the performance of David&Skene is not as good as majority vote.

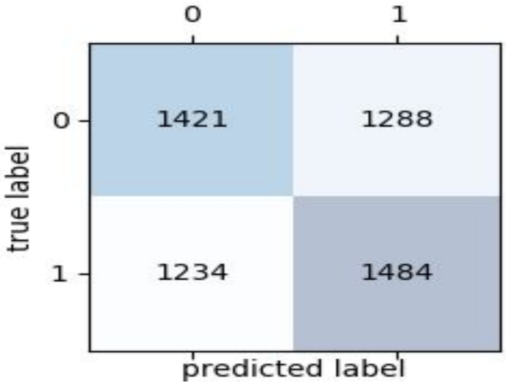

Figure 3.2 Confusion matrix of majority vote

The confusion matrix of decision tree is as figure 3.3: It is clear that TP is 1564, TN is 1300, which means this model make the number of right prediction is 1564+1300 = 2864; The FP is 1146 and NP is 1418=2564, which means the model makes wrong prediction is 1288 + 1234 = 2522; F1-score = 2*(precision * recall)/precision + recall = 0.54;

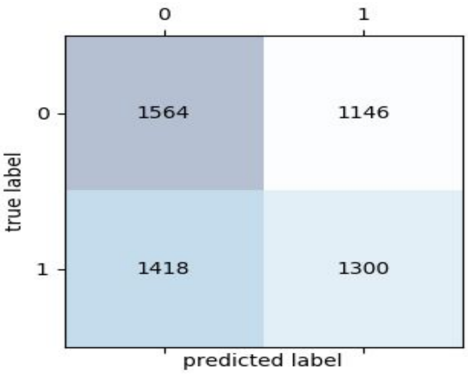

Figure 3.3 Confusion matrix of majority vote

The performance of three models: in terms of the dataset we have chosen, majority model performances best, followed by decision tree. David&Skene model doesn't performance as good as we expected. (Note: it's also depends on the sample we selected)

	Accuracy	F1-score	Rank
Decision Tree with original dataset	52.7634487841	0.503485670023	2
Decision Tree with majority vote	53.528653031	0.540619307832	1
Decision Tree with David & Skene	50.5988575640317	0.5360789063851877	3

Reference:

[1]	https://baike.baidu.com/item/%E5%86%B3%E7%AD%96%E6%A0%91/10377049?fr=aladdin decision tree
[2]	https://pandas.pydata.org/ how to use pandas
[3]	https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.groupby.html Group by in pandas:
[4]	https://www.geeksforgeeks.org/python-pandas-dataframe-drop_duplicates/ drop duplicated:
[5]	https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.unstack.html how to use unstack
[6]	https://www.jb51.net/article/150230.htm delete nan value
[7]	https://docs.pymc.io/notebooks/dawid-skene.html David&Skene