Introduction to Data Science

Text Classification

Course Code: SWE 336

Submitted To,

Ms Sayma Sultana Chowdhury

Assistant Professor

IICT, SUST

Submitted By,

Gourab Saha

Reg: 2017831004

Session: 2017-18

SWE, IICT, SUST

February 13, 2021

Assignment

Problem:

1. **Train five separate ML Classification models on this data and provide the classification report for each. Models: KNN, Naive Bayes, Random Forest, Decision Tree and ANN.**
2. **Why does it happen that the model gives very low f1 score for some classes but not the same for others?**
3. **Can you fix the low f1 score issue?**

Answer:

Answer to Question No: 1

1 Naive Bayes

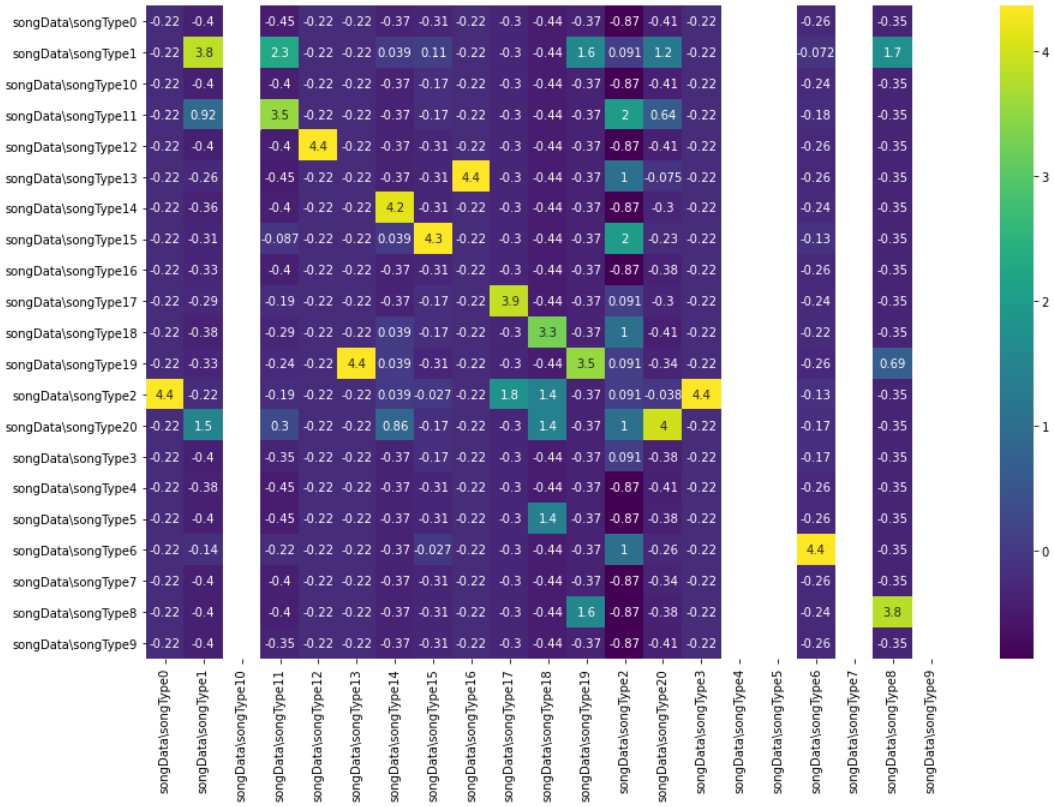
Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

The accuracy, macro average and weighted average are given below(Naive Bayes Model).

Cross Accuracy: 0.57 (+/- 0.03)

	precision	recall	f1-score	support
0	0.00	0.00	0.00	4
1	0.50	0.63	0.55	364
2	0.00	0.00	0.00	7
3	0.43	0.52	0.47	254
4	0.50	0.50	0.50	2
5	0.38	0.17	0.23	18
6	0.50	0.36	0.42	22
7	0.82	0.52	0.63	64
8	0.00	0.00	0.00	5
9	0.17	0.06	0.08	18
10	0.25	0.22	0.24	9
11	0.43	0.17	0.24	18
12	0.33	0.07	0.11	45
13	0.60	0.50	0.54	240
14	0.67	0.10	0.17	21
15	0.00	0.00	0.00	1
16	0.00	0.00	0.00	3
17	0.82	0.93	0.87	246
18	0.00	0.00	0.00	1
19	0.67	0.46	0.55	13
20	0.00	0.00	0.00	0
accuracy			0.57	1355
macro avg	0.34	0.25	0.27	1355
weighted avg	0.56	0.57	0.55	1355

Naive Bayes Model's Confusion Matrix.



2 KNN

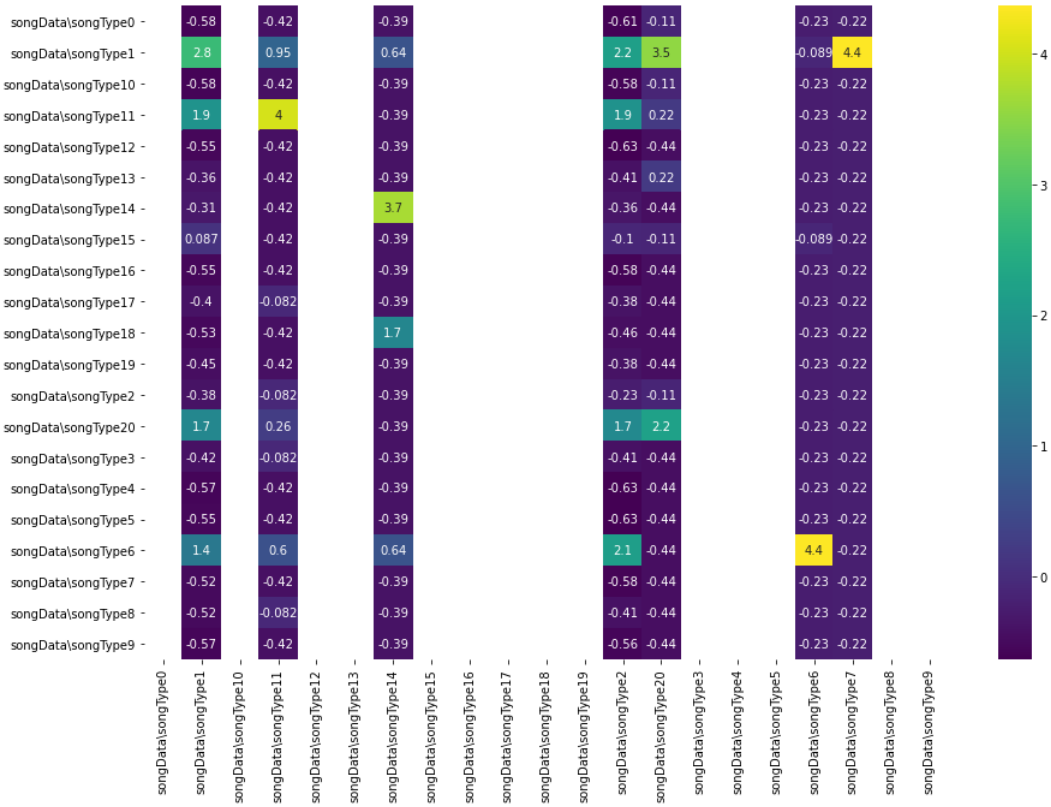
K-Nearest Neighbors (KNN) is one of the simplest algorithms used in Machine Learning for regression and classification problem. KNN algorithms use data and classify new data points based on similarity measures (e.g. distance function). Classification is done by a majority vote to its neighbors.

The accuracy, macro average and weighted average are given below(KNN Model).

Cross Accuracy: 0.18 (+/- 0.12)

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	0.27	0.60	0.38	331
2	0.00	0.00	0.00	3
3	0.50	0.05	0.09	265
4	0.00	0.00	0.00	2
5	0.00	0.00	0.00	24
6	0.50	0.13	0.21	31
7	0.00	0.00	0.00	63
8	0.00	0.00	0.00	4
9	0.00	0.00	0.00	22
10	0.00	0.00	0.00	12
11	0.00	0.00	0.00	18
12	0.03	0.53	0.06	30
13	0.29	0.03	0.06	238
14	0.00	0.00	0.00	20
15	0.00	0.00	0.00	1
16	0.00	0.00	0.00	2
17	0.94	0.12	0.22	263
18	0.00	0.00	0.00	6
19	0.00	0.00	0.00	14
20	0.00	0.00	0.00	4
accuracy			0.20	1355
macro avg	0.12	0.07	0.05	1355
weighted avg	0.41	0.20	0.17	1355

KNN Model's Confusion Matrix.



3 Decision Tree

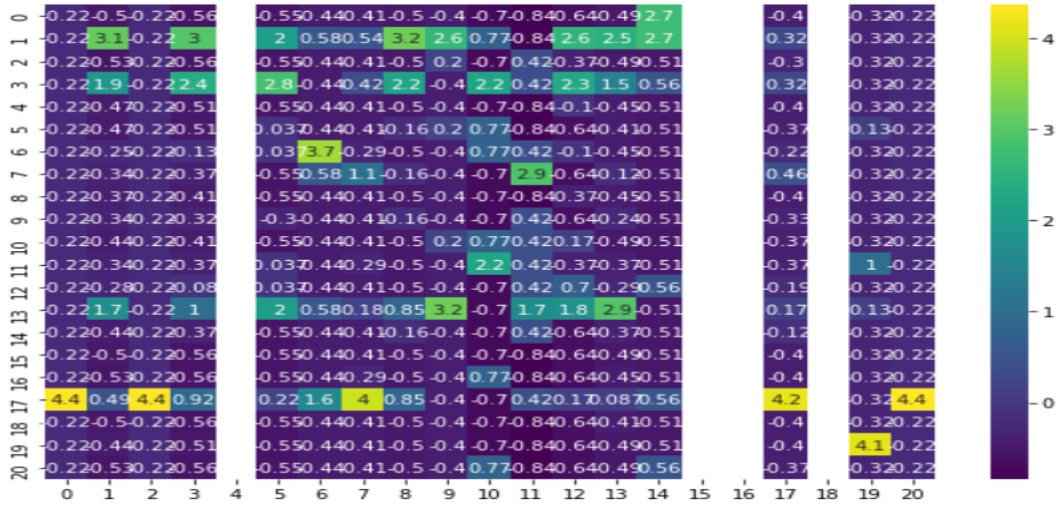
Decision Trees are a non-parametric supervised learning method used for both classification and regression tasks. Tree models where the target variable can take a discrete set of values are called classification trees.

The accuracy, macro average and weighted average are given below(Decision Tree Model).

Cross Accuracy: 0.33 (+/- 0.04)

	precision	recall	f1-score	support
0	0.00	0.00	0.00	4
1	0.33	0.35	0.34	332
2	0.00	0.00	0.00	6
3	0.25	0.25	0.25	249
4	0.00	0.00	0.00	6
5	0.04	0.17	0.07	12
6	0.44	0.11	0.18	35
7	0.18	0.21	0.19	61
8	0.00	0.00	0.00	10
9	0.00	0.00	0.00	22
10	0.10	0.08	0.09	13
11	0.07	0.04	0.05	24
12	0.10	0.13	0.11	38
13	0.33	0.34	0.34	241
14	0.00	0.00	0.00	20
15	0.00	0.00	0.00	1
16	0.00	0.00	0.00	3
17	0.54	0.49	0.52	258
18	0.00	0.00	0.00	3
19	0.67	0.71	0.69	14
20	0.00	0.00	0.00	3
accuracy			0.31	1355
macro avg	0.15	0.14	0.13	1355
weighted avg	0.32	0.31	0.31	1355

Decision Tree Model's Confusion Matrix.



4 Random Forest

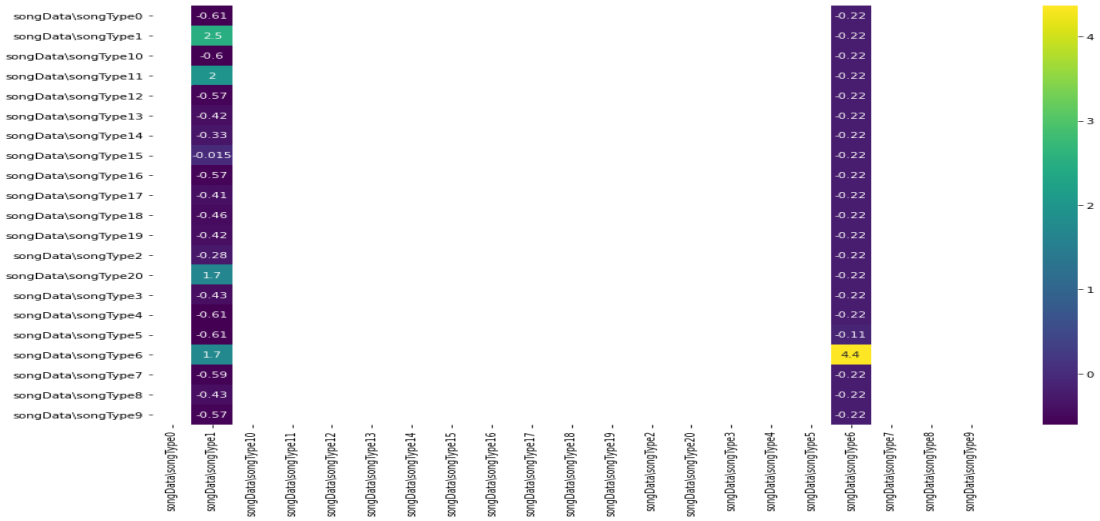
Random forest is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most used algorithms, because of its simplicity and diversity (it can be used for both classification and regression tasks).

The accuracy, macro average and weighted average are given below(Random Forest Model).

Cross Accuracy: 0.28 (+/- 0.03)

	precision	recall	f1-score	support
0	0.00	0.00	0.00	1
1	0.24	1.00	0.39	318
2	0.00	0.00	0.00	2
3	0.00	0.00	0.00	263
4	0.00	0.00	0.00	5
5	0.00	0.00	0.00	20
6	0.00	0.00	0.00	29
7	0.00	0.00	0.00	61
8	0.00	0.00	0.00	5
9	0.00	0.00	0.00	21
10	0.00	0.00	0.00	16
11	0.00	0.00	0.00	20
12	0.00	0.00	0.00	34
13	0.00	0.00	0.00	233
14	0.00	0.00	0.00	19
15	0.00	0.00	0.00	1
16	0.00	0.00	0.00	2
17	0.98	0.14	0.25	278
18	0.00	0.00	0.00	3
19	0.00	0.00	0.00	19
20	0.00	0.00	0.00	5
accuracy			0.26	1355
macro avg	0.06	0.05	0.03	1355
weighted avg	0.26	0.26	0.14	1355

Random Forest Model's Confusion Matrix.



5 Artificial Neural Network

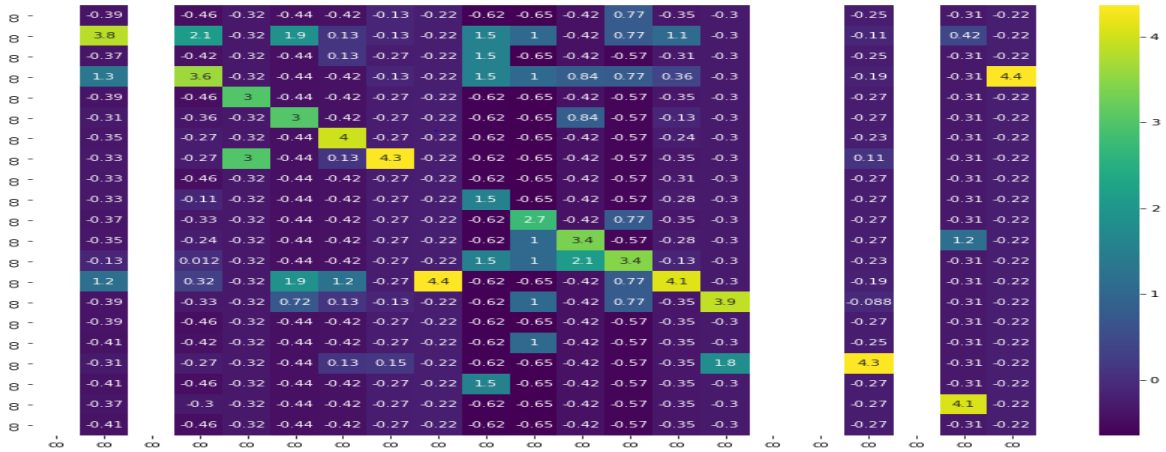
An artificial neural network (ANN) is the piece of a computing system designed to simulate the way the human brain analyzes and processes information. It is the foundation of artificial intelligence (AI) and solves problems that would prove impossible or difficult by human or statistical standards.

The accuracy, macro average and weighted average are given below(Artificial Neural Network).

Cross Accuracy: 0.53 (+/- 0.02)

	precision	recall	f1-score	support
0	0.00	0.00	0.00	3
1	0.47	0.61	0.53	344
2	0.00	0.00	0.00	4
3	0.45	0.53	0.49	258
4	0.00	0.00	0.00	2
5	0.50	0.05	0.09	21
6	0.86	0.22	0.35	27
7	0.89	0.22	0.35	78
8	0.00	0.00	0.00	10
9	1.00	0.12	0.21	17
10	0.00	0.00	0.00	13
11	0.38	0.13	0.19	23
12	0.00	0.00	0.00	37
13	0.52	0.58	0.55	224
14	0.67	0.11	0.19	18
16	0.00	0.00	0.00	1
17	0.77	0.95	0.85	246
18	0.00	0.00	0.00	3
19	1.00	0.15	0.26	20
20	0.00	0.00	0.00	6
accuracy			0.55	1355
macro avg	0.37	0.18	0.20	1355
weighted avg	0.55	0.55	0.51	1355

Artificial Neural Network's Confusion Matrix.



Answer to Question No: 2

The equation for calculating F1 score is:

$$2 * \frac{precision * recall}{precision + recall}$$

So F1 is high when precision and recall both are high.If precision or recall is low then F1 score will be low. Now, we know precision:

$$\frac{TruePositive}{TruePositive + FalsePositive}$$

The precision value will be high if False positive is low and precision will be low if False positive is high and vice versa. If False positive is low for a specific song type that means the model doesn't say other type to be this type. So model perfectly predict this song type.

If False positive is high for a specific class or category then model prediction is not good for this category. It says other category to be this category. Recall:

$$\frac{TruePositive}{TruePositive + FalseNegative}$$

Recall will be high when False negative is low and recall will be low when False negative is high. False negative low means most of the time model can predict correctly a specific category. And few times it says a specific category to other category. If False negative is high then model can't predict a specific category correctly.
So, F1 score is high when False positive and False negative bot is low.

Answer to Question No: 3

If the F1-score is the figure of merit, I would try to tune the class weights. It should be pretty easy, if we have a binary classification problem. We can feed the class weight a dictionary with the weights for each class. Here's a little example.

$clf = RandomForestClassifier()$
params = {'class weight': [{0:neg weight, 1:1} for neg weight in np.arange(1.0, 5.0, 0.5)]}
gs = GridSearchCV(estimator= clf, param grid = params, cv = 5)
gs.fit X train, y train