GROUP 19 : 3-D PRINTER MATERIAL PREDICTION

PROJECT REPORT BY:
KPS SHIVRATNA - 19BLC1012
PREETAL NILANG SHAH - 19BLC1013

1 INTRODUCTION

1.1 Overview

The problem statement given was to predict the material for 3D printer based on given parameters.3D printing materials are usually called by their traditional names such as ABS, nylon and etc. They are available in majority, but we have to be aware that many of the 3D printing materials only mimic true thermoplastics. We need to choose the right material to get a better printed object. Choosing the right material allows us to improve the shape, quality and function of our 3D printed part. Hence, selection of the correct 3D printing material is highly essential. To identify the type of material required after a 3D model is designed is a complicated task.

The aim of the study is to determine the material which will be perfectly suitable for the given use case. We have a dataset in which there are eleven setting parameters and one output parameters. Based on these input parameters we have to predict the best material for model. This model will predict whether to use ABS or PLA.

1.2 Purpose

The purpose of our project is to provide an output to the users based on the input parameters. The output will tell the users, which material is best suitable for their needs. We will use machine learning techniques to classify the materials as ABS or PLA based on the input parameters.

We are building a IBM Watson AutoAI Machine Learning to predict the material. We are developing a web application which is built using node red service. We make use of the scoring end point to give user input values to the deployed model. The model prediction is then showcased on User Interface. This model is to predict the best material to be used for building 3D models.

2 LITERATURE SURVEY

2.1 Existing problem

3D Printing gives us the ability to handle any level of complexity and ensures pinpoint accuracy. The things we can make are limited only by our imaginations. And it delivers the promise of mass customization. But this technology was supposed to revolutionize manufacturing. Putting creativity in everyone's hands, disrupts the production model.

1. Output/Quality Problems with 3D Printing.

In some ways, this is the most basic thing, but there are many quality-related problems with 3D printing today:

- Fragile, delaminated FDM (fused deposition modeling) parts
- Low-resolution output
- Materials

Now, to be fair, the materials are defifined by what can be extruded, squirted, or melted, but this is not based on their application or fifinal use. And even though there are some examples of multimaterials, it's typically only two at a time. So we're constraining ourselves.

2. The Process Is Unreliable.

The complexity of just getting the process to work is often daunting, and it involves too much fifiddling with formats, parameters, and mechanical adjustments.

3. The Workflflow.

The workflflow is old and outdated, and it's still based on the classic linear approach:

- Human: Design
- Computer: Document and Analyze

The 3D-printing workflflow usually doesn't take advantage of generative design or other recent breakthroughs. The problem with the current workflflow is that, fifirst, designers are drawing stuff, and then the robot is drawing stuff again in the 3D printer.

4. The Target: It's Wrong.

The fourth lamentation is that people have been aiming at the wrong target with their 3D-printing efforts. They've been happy creating prototypes, replacement parts, and trinkets; but what they should be focused on are fifinal parts and creating novel solutions to higher-level problems. In order to reach this new target, it's important to look at 3D printing holistically, through four facets of additive manufacturing: parts, system, materials, and process.

5: The Market: It's Prematurely Mature.

My last 3D-printing lamentation is that 3D-printer manufacturers seem to be prematurely solidifying standards and stiflfling innovation. Today, 3D-printing's problem is that the market remains "prematurely mature." Manufacturers are unfortunately mistaking that smaller, earlier chasm for the bigger one ahead. Customers are the enthusiasts, not the majority. Yet manufacturers are using business models meant to optimize later phases as if there were already a printer in every home.

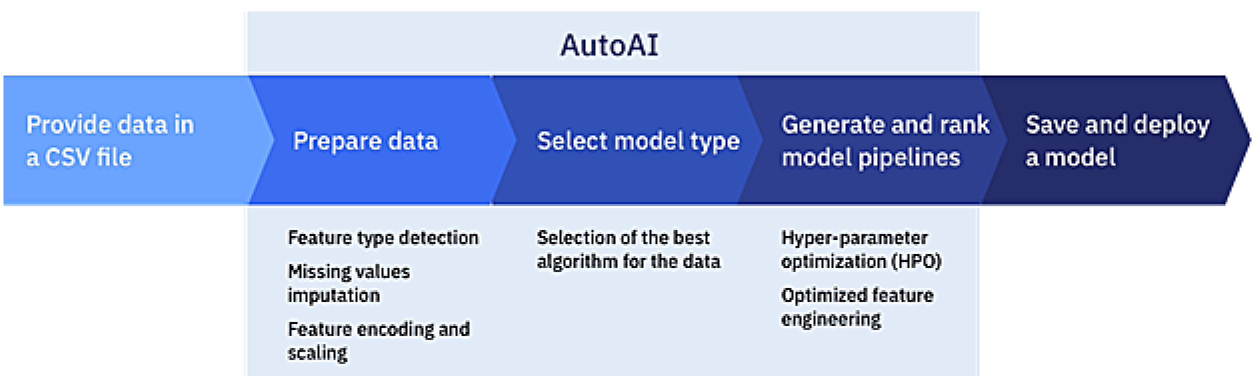
2.2 Proposed solution

This project prevents the people from choosing false material for 3D Printing. The model gets the appropriate data from the predictions done. After that we want to process those data using a suitable algorithm, then our model display which material to be used. To predict the appropriate material we are applying various machine learning algorithms.

The **k-nearest neighbors** algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

3 THEORETICAL ANALYSIS

3.1 Block diagram



3.2 Software designing

For Auto AI solution:

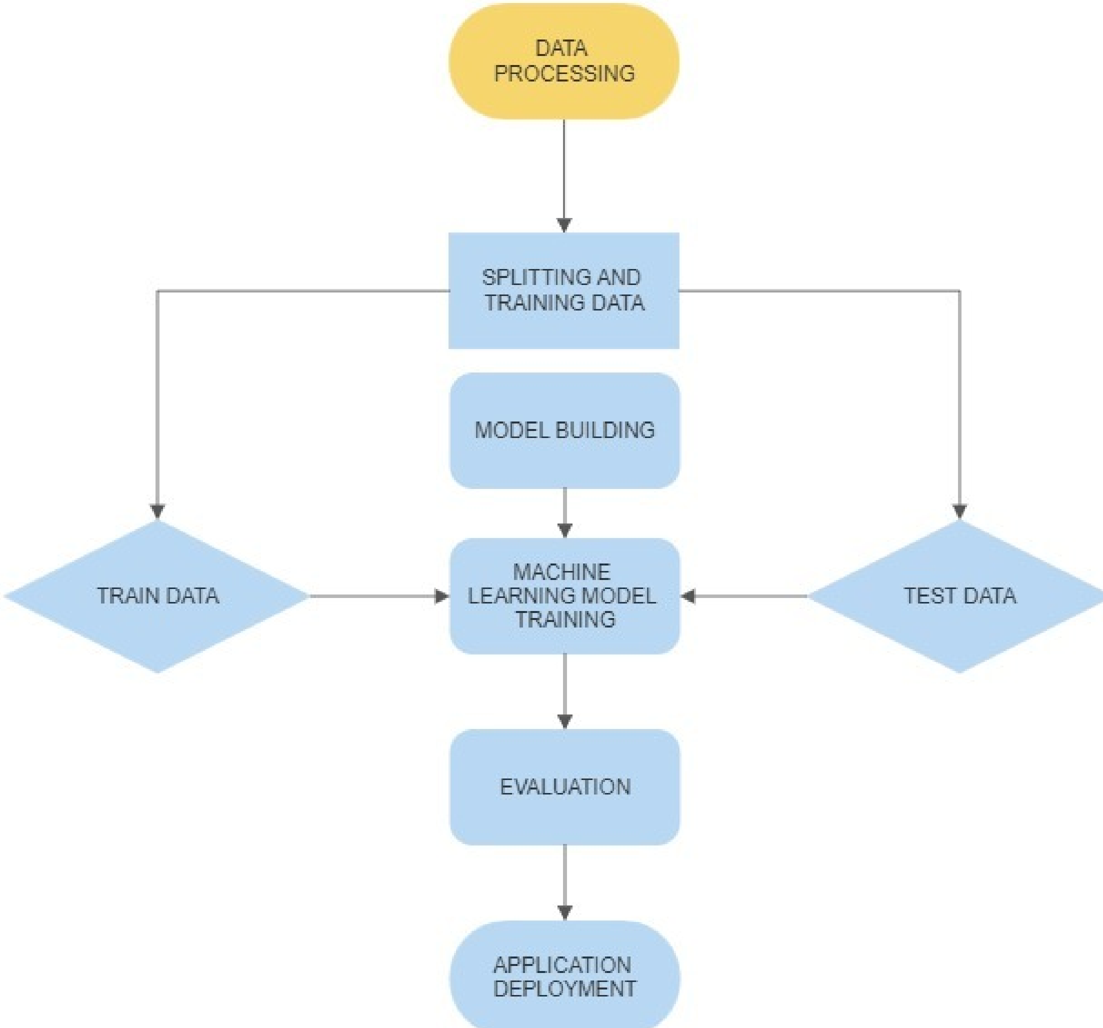
- Strategy: matching the problem with the solution.
- Dataset preparation and pre-processing. Data collection.
- Adding Dataset to the Watson Machine Learning.
- Doing Auto AI analysis to find out the best model.
- Model deployment.
- Making Node Red flow.
- Deploying the machine learning model through that Flow Application. For own ipynb Notebook solution:
- Strategy: matching the problem with the solution.
- Dataset preparation and pre-processing. Data collection. Data visualization. Labelling. Data selection. Data pre-processing. Data transformation.
- Dataset splitting into train data and test data.
- Modelling. Model training. Model evaluation and testing. Improving predictions with ensemble methods.
- Model deployment.
- Making Node Red flow.
- Deploying the machine learning model through that Flow Application.

4 EXPERIMENTAL INVESTIGATIONS

PARAMETERS :

1. layer height
2. wall thickness
3. infifill density
4. infifill pattern
5. nozzle temperature
6. bed temperature
7. print speed
8. material
9. fan speed
10. roughness
11. tension strength
12. elongation

5 FLOWCHART


DATA PROCESSING
SPLITTING AND TRAINING DATA
MODEL BUILDING
TRAIN DATA
MACHINE LEARNING MODEL TRAINING
TEST DATA
EVALUATION
APPLICATION DEPLOYMENT

6 RESULT

The model is able to predict appropriate material with good accuracy for provided parameters.

Training and Testing the Model

In [48]:
```
#Fitting Classifier to the Training set
clf = neighbors.KNeighborsClassifier()
clf.fit(x_train,y_train)
```

Out[48]:
```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
                     metric_params=None, n_jobs=None, n_neighbors=5, p=2,
                     weights='uniform')
```

In [49]:
```
#Predicting the Test Set results
y_pred=clf.predict(x_test)
```

In [50]:
```
y_pred
```

Out[50]:
```
array([0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1,
       1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0,
       1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0,
       0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0])
```

In [52]:
```
#Accuracy score on Test and Train
from sklearn.metrics import accuracy_score, recall_score, roc_auc_score, confusion_matrix

print("\nAccuracy score: %f" %(accuracy_score(y_test,y_pred) * 100))
print("Recall score : %f" %(recall_score(y_test, y_pred) * 100))
print("ROC score : %f\n" %(roc_auc_score(y_test, y_pred) * 100))
print(confusion_matrix(y_test, y_pred))
```

```
Accuracy score: 93.827160
Recall score : 90.625000
ROC score : 93.271684

[[47  2]
 [ 3 29]]
```

Predictions for new data

In [1]:
```
import os
os.chdir("C:/Users/parents gift/Desktop/deployment-flask")
```

In [2]:
```
from sklearn import model_selection, neighbors
import pickle
```

In [3]:
```
model = pickle.load(open('model.pkl','rb'))
```

In [4]:
```
n1=[[24,15,1.2]]
```

In [5]:
```
p1=model.predict(n1)
if(p1[0]==0):
    print("Material is abs")
else:
    print("Material is pla")
```

```
Material is pla
```

In [6]:
```
n2=[[56,25,1.6]]
```

In [7]:
```
p1=model.predict(n2)
if(p1[0]==0):
    print("Material is abs")
else:
    print("Material is pla")
```

```
Material is pla
```

In [8]:
```
n3=[[26,17,1.1]]
```

In [9]:
```
p1=model.predict(n3)
if(p1[0]==0):
    print("Material is abs")
else:
    print("Material is pla")
```

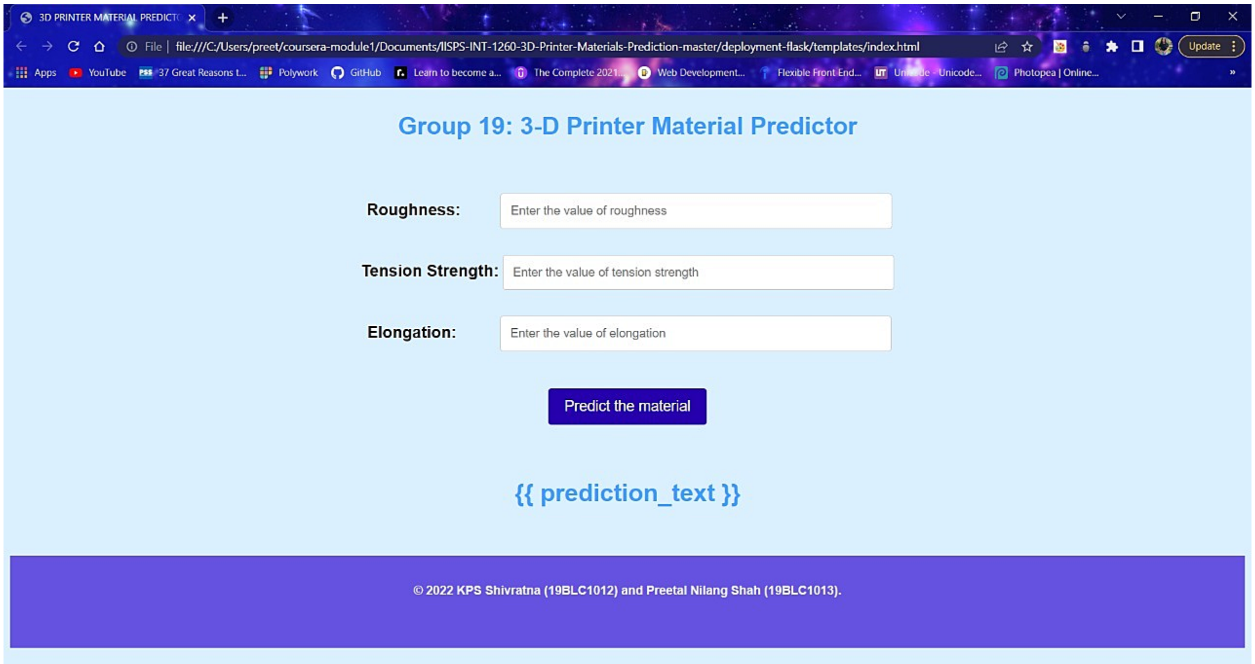
```
Material is abs
```

In [10]:
```
n4=[[90,6,0.8]]
```

In [11]:
```
p1=model.predict(n4)
if(p1[0]==0):
    print("Material is abs")
else:
    print("Material is pla")
```

```
Material is abs
```

In [ ]:



7 ADVANTAGES & DISADVANTAGES

Advantages

1. Speed: One of the biggest advantages of 3D printing technology is Rapid Prototyping.
2. Cost: For small production runs and applications, 3D printing is the most cost-effective manufacturing process.
3. Flexibility.
4. Competitive Advantage
5. Tangible Design and Product Testing.
6. Quality.
7. Consistency.
8. Risk Reduction.

Disadvantages

● User should have the idea on all the parameters and units of each parameter.

8 APPLICATIONS

A growing range of 3D printer and 3D printing material options has widened the potential applications of 3D printing significantly. From 3D printed airplane parts to medical devices, 3D printing has made it easier than ever to create lighter, more effective parts at a faster rate (and a lower cost) than traditional methods. Industries using 3D printing include:

- – Aerospace
- – Automotive
- – Medical
- – Dental
- – Consumer Goods
- – Manufacturing Common applications for 3D printing include:
- – Rapid prototyping
- – Modeling
- – Low-volume, custom parts
- – Production parts
- – Jigs and fixtures

9 CONCLUSION

Since anyone who is getting a part 3D printed does not want to waste resources and wants to obtain a reliable product. Our application helps in predicting the best material for 3d printing their object based on the past data.

Based on the 11 inputs entered by the user, the model predicts the best material for 3D printing an object. And gives the output according to the entries in the Node red application. Unlike traditional methods there is no wastage of test samples. Higher accuracy can reduces errors in wrong selection of material. Reduce the cost of finding out best material for 3D printing an object. Easy user interface with straight forward prediction. The model is limited to predict the material for only those materials which have exactly 11 compositions in their mixture. The input parameters need to be correctly examined before the prediction is made. It can be used to predict the best material suitable for 3D printing an object that is made using several parameters. It can also be made into a phone app. With this model now engineers would be able to determine the best material for 3D printing an object. Based on this many would be able to advise which material to use for 3D printing an object based on the given input parameters. This model can predict the outcome with many different inputs within seconds. The model will save a lot of time. Experiment cost is also reduced which creates a bigger opportunity in cost effectiveness work.

10 FUTURE SCOPE

While the fundamentals of its function may remain the same, we can expect to see more accurate machines, complex materials, and exciting applications come from 3D printing in the future.

11 BIBILOGRAPHY

1. https://en.wikipedia.org/wiki/3D_printing
2. https://www.cmac.com.au/blog/top-10-materials-used-industrial-3d-printing
3. https://www.cati.com/blog/2019/01/how-3d-printing-works

APPENDIX

Model.py :

```
# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import pickle

dataset = pd.read_csv('3d printer modified.csv')

dataset['roughness'].fillna(dataset['roughness'].mean(), inplace=True)

dataset['tension_strenght'].fillna(dataset['tension_strenght'].mean(), inplace=True)

dataset['elongation'].fillna(dataset['elongation'].mean(), inplace=True)

material = {'abs': 0, 'pla': 1}
dataset.material = [material[item] for item in dataset.material]

X = dataset.loc[:, ['roughness','tension_strenght','elongation']]
from sklearn import preprocessing
minmax=preprocessing.MinMaxScaler(feature_range=(0,1))
minmax.fit(X).transform(X)
y = dataset.iloc[:, -1]
```

```
#Splitting Training and Test Set
#Since we have a very small dataset, we will train our model with all availabe data.

from sklearn import model_selection, neighbors
clf = neighbors.KNeighborsClassifier()

#Fitting model with trainig data
clf.fit(X,y)

# Saving model to disk
pickle.dump(clf, open('model.pkl','wb'))

# Loading model to compare the results
model = pickle.load(open('model.pkl','rb'))
print(model.predict([[21, 14, 1.5]]))
```

App.py :

```
import numpy as np
from flask import Flask, request, jsonify, render_template
import pickle

app = Flask(__name__)
model = pickle.load(open('model.pkl', 'rb'))

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/predict',methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    int_features = [float(x) for x in request.form.values()]
    final_features = [np.array(int_features)]
    prediction = model.predict(final_features)
```

```
    output = prediction[0]
    if(output==0):
        bar='Material is abs'
    else:
        bar='Material is pla'

    return render_template('index.html', prediction_text=bar)

@app.route('/predict_api',methods=['POST'])
def predict_api():
    '''
    For direct API calls trought request
    '''
    data = request.get_json(force=True)
    prediction = model.predict([np.array(list(data.values()))])

    output = prediction[0]
    if(output==0):
        bar='Material is abs'
    else:
        bar='Material is pla'

    return jsonify(bar)


if __name__ == "__main__":
    app.run(debug=True)
```

12 PROJECT DEMO VIDEO LINK

https://drive.google.com/file/d/1hr8gj1-uB1-h3cUw3CJRTfhN0vKMhPQ2/view?usp=sharing