# MACHINE LEARNING PROJECT

## QUESTION PAPER ANALYSER


- CREATED BY :
  - RINKU AHIRWAR (20103124)
  - SAKSHAM KARNATAK (20103127)
  - SNEHAL DHAWAN (20103142)


In [1]:

```
# IMPORTING LIBRARIES
import pandas as pd
import numpy as np
```

In [2]:

```
#IMPORTING THE DATA SET AND REMOVING ANY NULL VALUES
df=pd.read_csv("math_dataset.csv",encoding='latin1')
df.dropna(how='any',inplace=True)
```

In [3]:

```
df.head()
```

Out[3]:

| | problem | type |
|---|---|---|
| 0 | An international meeting is held between Engla... | Counting & Probability |
| 1 | Express $\frac{6!+4!}{5!}$ as a mixed number. | Counting & Probability |
| 2 | What is the shortest distance that can be trav... | Counting & Probability |
| 3 | How many paths are there from $A$ to $B$ on th... | Counting & Probability |
| 4 | Krishanu and Shaunak each pick an integer at r... | Counting & Probability |

In [4]:

```
df.shape
```

Out[4]:

```
(1399, 2)
```

In [5]:

```
df.columns
```

Out[5]:

```
Index(['problem', 'type'], dtype='object')
```

In [6]:

```
# SETTING THE TARGET FUNCTION INTO X
x=df.iloc[:,1]
x
```

Out[6]:

```
0       Counting & Probability
1       Counting & Probability
2       Counting & Probability
3       Counting & Probability
4       Counting & Probability
                ...
1395              Precalculus
1396              Precalculus
1397              Precalculus
1398              Precalculus
1399              Precalculus
Name: type, Length: 1399, dtype: object
```

In [7]:

```
# ASSIGNING THE VALUES FROM 1 TO 7 TO EACH TOPIC
df.type.replace({'Counting & Probability':1,'Algebra':2,'Prealgebra':3,
                 'Geometry':4,'Precalculus':5,'Number Theory':6,'Intermediate Algebra':7})
```

Out[7]:

```
0       1
1       1
2       1
3       1
4       1
       ..
1395    5
1396    5
1397    5
1398    5
1399    5
Name: type, Length: 1399, dtype: int64
```

In [8]:

```
# SPLITTING THE DATA INTO TRAINING AND TESTING
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(df.problem,df.type,test_size=0.25)
```

In [9]:

```
# CONVERTING THE STRING QUESTIOS INTO NUMBER MATRIX
from sklearn.feature_extraction.text import CountVectorizer
v=CountVectorizer()
x_train_count=v.fit_transform(x_train.values)
x_test_count=v.transform(x_test)
x_test_count.toarray()[:]
```

Out[9]:

```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]], dtype=int64)
```

In [10]:

```
# PREPARING MODEL AND TRAINING IT
from sklearn.naive_bayes import MultinomialNB
model=MultinomialNB()
model.fit(x_train_count,y_train)
```

Out[10]:

```
MultinomialNB()
```

In [11]:

```
# GENERATING INPUT
a=['find the area of a square with side 8 cm.',
  'find the probaility of head in fliping a coin',
  'solve 3x^2 + 2x',
  'find the probability of drawing a card which is club',
  'What value of x^2 will give the minimum value for x^2- 14x + 3 ?',
  'Find the minimum value of the function f(x) = sqrt{-x^2 + 4x + 21} - sqrt{-x^2 + 3x + 10
  'What is the sum of the three digit cubes that are the cubes of either squares or cubes?'
  'In how many ways can the letters of the word ''COPYRIGHT'' be arranged?',
  'Evaluate d/dx cos 99 ']
```

In [12]:

```
# PREDICTING THE INPUT
k=v.transform(a)
r=model.predict(k).tolist()
s=r
s
```

Out[12]:

```
['Geometry',
 'Algebra',
 'Algebra',
 'Counting & Probability',
 'Algebra',
 'Intermediate Algebra',
 'Number Theory',
 'Counting & Probability',
 'Precalculus']
```

In [13]:

```
# ASSIGINING THE CHAPTER VALUES TO THE CHAPTERS
j=0;
for i in r:
    if i=='Counting & Probability':
        s[j]=1
    elif i=='Algebra':
        s[j]=2
    elif i=='Prealgebra':
        s[j]=3
    elif i=='Geometry':
        s[j]=4
    elif i=='Precalculus':
        s[j]=5
    elif i=='Number Theory':
        s[j]=6
    elif i=='Intermediate Algebra':
        s[j]=7
    j=j+1;


s
```

Out[13]:

```
[4, 2, 2, 1, 2, 7, 6, 1, 5]
```

In [14]:

```
# COUNTING FOR NUMBER OF QUESTIONS FROM EACH CHAPTER
chap=[1,2,3,4,5,6,7]
j=0
for i in chap:
    chap[j]=s.count(i)
    j=j+1
chap
```

Out[14]:

```
[2, 3, 0, 1, 1, 1, 1]
```

In [15]:

```
model.score(x_test_count,y_test)
```

Out[15]:

```
0.8236415633937083
```

In [16]:

```
# CALCULATING CHAPTER WISE WEIGHTAGE OF THE PREDICTED RESULT
sum=0
for i in chap :
    sum+=i

p=0
for i in chap :

    chap[p]=(i/sum)*100
    p=p+1

topic={1:'Counting & Probability',2:'Algebra',3:'Prealgebra',4:'Geometry',5:'Precalculus',6
print('CHAPTER   WEIGHTAGE(%)  ')
for i in range(1,8):
    print(topic[i],'\t',chap[i-1])
```

```
CHAPTER   WEIGHTAGE(%)
Counting & Probability   22.22222222222222
Algebra          33.33333333333333
Prealgebra       0.0
Geometry         11.11111111111111
Precalculus      11.11111111111111
Number Theory    11.11111111111111
Intermediate Algebra     11.11111111111111
```