In [1]:
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB, MultinomialNB
from sklearn.metrics import accuracy_score, classification_report
#import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.feature_extraction.text import CountVectorizer
```

In [2]:
```
df_train_irre =pd.read_csv(r'C:\Users\oumar\Desktop\Python_odev\ddi_nb_bonus\haiti\train\irrelevant', sep='|',names=['sentences']
df_train_irre['class']= 0

df_train_re =pd.read_csv(r'C:\Users\oumar\Desktop\Python_odev\ddi_nb_bonus\haiti\train\relevant', sep='|',names=['sentences'])
df_train_re['class']= 1
```

In [3]:
```
df_merged = df_train_irre.append(df_train_re, ignore_index=True).sample(frac=1)
X_train=df_merged["sentences"].to_numpy()
y_train=df_merged["class"].to_numpy()

df_merged.head()
```

```
C:\Users\oumar\AppData\Local\Temp\ipykernel_11712\3629566172.py:1: FutureWarning: The frame.append method is deprecated and wil
l be removed from pandas in a future version. Use pandas.concat instead.
  df_merged = df_train_irre.append(df_train_re, ignore_index=True).sample(frac=1)
```

Out[3]:

	sentences	class
820	i have a child who 's father is in a foreign c...	1
1055	i have a lot of problems and i do not have any...	1
697	where i could find disinfecting to aspeger the...	1
1207	could you give help by giving some portable to...	1
453	we are dying of hunger . we are located at sou...	1

In [4]:
```
y_train
```

Out[4]:
```
array([1, 1, 1, ..., 1, 0, 0], dtype=int64)
```

In [5]:
```
df_test_re =pd.read_csv(r'C:\Users\oumar\Desktop\Python_odev\ddi_nb_bonus\haiti\test\relevant', sep='|',names=['sentences'])
df_test_re['class']=1

df_test_irre =pd.read_csv(r'C:\Users\oumar\Desktop\Python_odev\ddi_nb_bonus\haiti\test\irrelevant', sep='|',names=['sentences'])
df_test_irre['class']=0
```

In [6]:
```
df_merged = df_test_irre.append(df_test_re, ignore_index=True).sample(frac=1)
X_test=df_merged["sentences"].to_numpy()
y_test=df_merged["class"].to_numpy()

df_merged.head()
```

```
C:\Users\oumar\AppData\Local\Temp\ipykernel_11712\2562285556.py:1: FutureWarning: The frame.append method is deprecated and wil
l be removed from pandas in a future version. Use pandas.concat instead.
  df_merged = df_test_irre.append(df_test_re, ignore_index=True).sample(frac=1)
```

Out[6]:

	sentences	class
16	this message is not very important to translate .	0
25	try business with digicel haiti	0
72	we need food distributions in larat kadet (to...	1
24	good evening!could you refer me to the ministe...	0
58	. .. right now , i left the city , becuase p...	1

In [7]:
```
vectorizer = CountVectorizer()
X_trainM = vectorizer.fit_transform(X_train).toarray()
X_testM = vectorizer.transform(X_test).toarray()

print(vectorizer.vocabulary_)
```

```
maged : 745, afraid : 162, 1hour : 24, trained : 2783, agronomy : 176, field : 1091, short : 2459, reopenning : 2280,
'march': 1696, '1st': 27, 'seguineau': 2413, 'corner': 680, 'bertholy': 376, '610': 94, 'review': 2318, 'lillavois47': 1621,
'29': 51, 'history': 1340, 'prepare': 2137, 'future': 1184, 'enjoy': 983, 'resistred': 2303, 'programm': 2165, 'pasket': 201
2, 'sharing': 2450, 'sell': 2419, 'enter': 989, 'difficulty': 835, 'ida': 1390, 'poupla': 2124, 'trofort': 2802, 'alone': 20
3, 'ave': 304, 'christophe': 566, 'imp': 1403, 'lavisse': 1583, 'muscular': 1838, 'springiness': 2566, 'extensibility': 1040,
'property': 2171, 'muscles': 1837, 'distort': 859, 'elongation': 965, 'usual': 2878, 'shape': 2448, 'injuries': 1433, 'compli
cation': 636, 'okt': 1942, 'ou': 1976, 'purples': 2194, 'disaster': 849, 'stricken': 2603, 'trans': 2785, 'borther': 417, 'ur
gence': 2866, 'technicien': 2676, 'informatique': 1426, 'worked': 3008, 'mois': 1799, 'distributors': 865, 'indicate': 1416,
'bottle': 422, 'martissan': 1707, 'dkayet': 873, 'diarrhea': 821, 'rampant': 2219, 'undermining': 2837, 'regarding': 2256, 'e
xpect': 1031, 'identified': 1394, 'metropolitan': 1757, 'repair': 2281, 'recieve': 2244, 'associations': 285, 'french': 1168,
'half': 1283, 'english': 980, 'mw': 1840, 'tarp': 2666, 'buid': 457, 'christ': 564, 'twelve': 2823, 'replies': 2284, 'below':
370, 'shower': 2462, 'tnh': 2751, 'physical': 2061, 'poop': 2108, 'themselve': 2710, 'cutoff': 733, 'face': 1046, 'agoman': 1
72, 'acireh': 132, 'action': 137, 'renovation': 2276, 'monseigneur': 1809, 'giyou': 1216, 'slelak': 2501, 'tn': 2750, 'reside
ntial': 2300, 'cloths': 600, 'enroll': 986, 'provide': 2176, 'nutrition': 1920, 'increse': 1413, 'beleive': 363, 'impossibl
e': 1406, 'donate': 893, 'labelaire': 1555, '43': 71, 'takes': 2656, 'function': 1180, 'ending': 978, 'mean': 1724, 'identif
y': 1395, 'medicin': 1738, 'resting': 2311, 'identification': 1393, 'choosing': 563, 'minotri': 1778, 'veau': 2889, 'shortag
e': 2460, 'pestel': 2051, 'throw': 2738, 'away': 310, 'garbage': 1189, 'closes': 596, 'seigneur': 2416, 'il': 1398, 'convinci
ng': 672, 'persuaded': 2049, 'messieurs': 1753, 'odors': 1930, 'themselves': 2711, 'bosses': 419, 'nice': 1886, 'talks': 266
1, 'starvin': 2576, 'aftershake': 165, 'attain': 293, 'level': 1609, 'long': 1644, '400': 68, 'demunis': 789, 'pak': 1992, 'j
oin': 1493, '41': 69, 'myrthe': 1842, 'faustin': 1066, 'concerns': 647, 'offer': 1934, 'ditribute': 869, 'goods': 1233, 'fest
```

In [8]:
```
# Function that handles sample splitting, model fitting and report printing
def mfunc(X_train, X_test, y_train, y_test, typ):

    # Create training and testing samples
    #X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

    # Fit the model
    model = typ
    clf = model.fit(X_train, y_train)

    # Predict class labels on a test data
    pred_labels = model.predict(X_test)

    # Print model attributes
    #print('Classes: ', clf.classes_) # class labels known to the classifier
    #if str(self)=='GaussianNB()':
    #    print('Class Priors: ',clf.class_prior_) # prior probability of each class.
    #else:
        #print('Class Log Priors: ',clf.class_log_prior_) # log prior probability of each class.

    # Use score method to get accuracy of the model
    print('----------------------------------------------------------')
    score = model.score(X_test, y_test)
    print('Accuracy Score: ', score)
    print('----------------------------------------------------------')

    # Look at classification report to evaluate the model
    print(classification_report(y_test, pred_labels))

    # Return relevant data for chart plotting
    return X_train, X_test, y_train, y_test, clf, pred_labels
```

In [9]:
```
# Fit the model and print the result
X_train1, X_test1, y_train1, y_test1, clf, pred_labels, = mfunc(X_trainM, X_testM, y_train, y_test, MultinomialNB())
```

```
----------------------------------------------------------
Accuracy Score:  0.8048780487804879
----------------------------------------------------------
              precision    recall  f1-score   support

           0       0.91      0.69      0.78        42
           1       0.74      0.93      0.82        40

    accuracy                           0.80        82
   macro avg       0.82      0.81      0.80        82
weighted avg       0.83      0.80      0.80        82
```

In [ ]:
```

```