Results and comparisons

```
df_accuracy_rishab.sort_values(by = 'Test Accuracy', ascending = False, inplace=True)
df_accuracy_rishab.reset_index(drop = True, inplace=True)
df_accuracy_rishab
```

Out[157]:

	Model	Test Accuracy	Training time (sec)
0	Decision Tree	0.892857	57.08
1	SVM	0.892857	5.17
2	Random Forest	0.875000	65.74
3	MultinomialNB	0.835714	16.83
4	KNN	0.832143	87.83

In [161]:
```
df_accuracy_sushant = pd.DataFrame(models_accuracy_sushant)
df_accuracy_sushant.columns = ['Model', 'Test Accuracy', 'Training time (sec)']
df_accuracy_sushant.sort_values(by = 'Test Accuracy', ascending = False, inplace=True)
df_accuracy_sushant.reset_index(drop = True, inplace=True)
df_accuracy_sushant
```

Out[161]:

	Model	Test Accuracy	Training time (sec)
0	Random Forest	0.952206	104.44
1	SVM	0.933824	10.84
2	Decision Tree	0.882353	134.36
3	MultinomialNB	0.863971	10.31
4	KNN	0.856618	132.72

In [165]:
```
df_accuracy_shruti = pd.DataFrame(models_accuracy_shruti)
df_accuracy_shruti.columns = ['Model', 'Test Accuracy', 'Training time (sec)']
df_accuracy_shruti.sort_values(by = 'Test Accuracy', ascending = False, inplace=True)
df_accuracy_shruti.reset_index(drop = True, inplace=True)
df_accuracy_shruti
```

Out[165]:

	Model	Test Accuracy	Training time (sec)
0	SVM	0.937729	11.67
1	Random Forest	0.919414	79.79
2	MultinomialNB	0.879121	7.38
3	KNN	0.879121	110.01
4	Decision Tree	0.871795	168.78

```
    }

    for name, model in models.items():
        start = perf_counter()
        model['model'].fit(dtv_rishab, y_train_rishab)
        duration = perf_counter() - start
        duration = round(duration,2)
        model["perf"] = duration
        print(f"{name:20} trained in {duration} sec in Rishab's dataset")
```

```
Random Forest        trained in 65.74 sec in Rishab's dataset
MultinomialNB        trained in 16.83 sec in Rishab's dataset
KNN                  trained in 87.83 sec in Rishab's dataset
Decision Tree        trained in 57.08 sec in Rishab's dataset
SVM                  trained in 5.17 sec in Rishab's dataset
Wall time: 4min 11s
```

In [158]:
```
for name, model in models.items():
    start = perf_counter()
    model['model'].fit(dtv_sushant, y_train_sushant)
    duration = perf_counter() - start
    duration = round(duration,2)
    model["perf"] = duration
    print(f"{name:20} trained in {duration} sec in Sushant's dataset")
```

```
Random Forest        trained in 104.44 sec in Sushant's dataset
MultinomialNB        trained in 10.31 sec in Sushant's dataset
KNN                  trained in 132.72 sec in Sushant's dataset
Decision Tree        trained in 134.36 sec in Sushant's dataset
SVM                  trained in 10.84 sec in Sushant's dataset
```

In [162]:
```
for name, model in models.items():
    start = perf_counter()
    model['model'].fit(dtv_shruti, y_train_shruti)
    duration = perf_counter() - start
    duration = round(duration,2)
    model["perf"] = duration
    print(f"{name:20} trained in {duration} sec in Shruti's dataset")
```

```
Random Forest        trained in 79.79 sec in Shruti's dataset
MultinomialNB        trained in 7.38 sec in Shruti's dataset
KNN                  trained in 110.01 sec in Shruti's dataset
Decision Tree        trained in 168.78 sec in Shruti's dataset
SVM                  trained in 11.67 sec in Shruti's dataset
```

Accuracy vs Model

Average accuracy vs Model

Time taken per Model vs Model