**Can you come up with a reason for the good performance? What would be the main practical problems we would face if we were not to make this assumption for this particular dataset?**

Naive Bayes is a simple and efficient classification algorithm. In problems like spam filtering, where features (words) are largely independent of each other, its “naive” assumption of feature independence leads to accurate predictions. Naive Bayes is known to be robust to irrelevant features and noisy data. In the case of spam filtering, where emails may contain a mix of relevant and irrelevant words, Naive Bayes can effectively filter out noise and focus on important features for classification. Because of these factors, Naive Bayes classifier is showing good performance. Relaxing the assumption of feature independence would lead to a more complex model, exacerbate data sparsity issues and lead to overfitting. Also, the model would become more sensitive to the distribution of training data and the presence of rare or unseen feature combinations. This could lead to poor generalization performance.

**How long did your classifier take to train and classify? Given this, how scalable do you think the Naïve Bayes classifier is to large datasets? Can you come up with a good reason for this?**

The classifier took 0.05 seconds to train and classify the dataset. This indicates that Naive Bayes classifier is highly scalable to large datasets. The simplicity, efficiency, linear time complexity, and parallelizability of Naive Bayes make it highly scalable to large datasets.

**How does Naïve Bayes compute the probability of an e-mail belonging to a class (spam/not spam)?**

Naive Bayes computes the probability of an email belonging to a class using Bayes theorem.
Mathematically, Bayes theorem is expressed as:

$$P(C/E) = \frac{P(E/C).P(C)}{P(E)}$$

Here, P(C/E) is the probability of the class C given the email E.
P(E/C) is probability of observing the email E given the class C.
P(C) is the prior probability of the class C.
P(E) is the total probability of obtaining the email E.

**Compute the conditional probability of observing the word “3d” given that an e-mail is spam P(3d|spam) and that it is non-spam P(3d|non-spam).**

```
word_freq_3d_binarized
  0                                2781.0 1775.0
  1                                   9.0   40.0
  [total]                          2790.0 1815.0
```

P(3d|spam) = $\frac{40.0}{1815.0}$ = 0.022

P(3d|non-spam) = $\frac{9.0}{2790.0}$ = 0.0032