BT3102: Computational Methods for Business Analytics

Team: Haireton

Chew Kai Ying, Claire	A0189716M
Elton Tay Jing Sheng	A0201534W
Mohamad Nurhakiem Bin Mohamd Rasid	A0190241R

BT3102 Project Qns 2-5

2c) Naive prediction accuracy: 65.31%

```
Naive prediction accuracy:     900/1378 = 0.6531204644412192
```

3a) $j^* = argmax_j P(y = j \mid x = w) = argmax_j \frac{P(y=j, x=w)}{P(x=w)}$.

Since $P(x = w)$ is constant for each word as we are finding argmax of the tag, we will only be evaluating $j^* = argmax_j P(y = j, x = w)$ by using the output probabilities from (2a) multiplied by $P(y = j)$ for each tag.

3c) Naive prediction 2 accuracy: 69.30%

```
Naive prediction2 accuracy:    955/1378 = 0.693033381712627
```

4c) Viterbi Algorithm accuracy: 75.76%

```
Viterbi prediction accuracy:   1044/1378 = 0.7576197387518142
```

5a) After analysing “twitter_train.txt”, we noticed linguistic patterns present in tweets. For example, the characters following “@USER_” and “http://” differ, but the predictive tags associated with them are generally the same. Therefore, we performed an automatic preprocessing of the data by clustering words following these patterns into common meaningful groups. We did this by reducing the relevant words to their respective reduced word (i.e. “@USER” or “http”) and associating each reduced word with the tag. This pattern can be confirmed with the naïve_output_probs.txt with a noticeably high frequency tag ‘@’ associated with ‘@user’ and tag ‘U’ associated with ‘http’.

With this, we managed to improve the accuracy of our Naive predictions as follows:

```
Naive prediction accuracy:     919/1378 = 0.6669085631349783
Naive prediction2 accuracy:    967/1378 = 0.7017416545718432
```

5c) Viterbi_predict2 accuracy: 75.62%

```
Viterbi2 prediction accuracy:  1042/1378 = 0.7561683599419449
```