Interpolation (bigram, unigram)	Add-K (prior, emission)	OOV cutoff	Accuracy
0.98 0.02	0.1 0.1	10	90.49%
0.98 0.02	0.1 0.1	5	92.05%
0.90 0.10	0.1 0.1	5	92.13%
0.98 0.02	0.05 0.05	5	92.16%
0.98 0.02	0.05 0.05	4	92.65%
0.98 0.02	0.05 0.05	3	93.29%
0.90 0.10	0.1 0.1	2	94.06%
0.98 0.02	0.05 0.05	2	94.21%
0.99 0.01	0.05 0.05	2	94.20%
0.99 0.01	0.05 0.05	1	95.70%

Table 1: Bigram HMM Tagger Accuracy

Question 1.1 Bigram HMM

In the bigram HMM model, we handled OOV symbols in the emission model by classifying it based on its word form into *MENTION*, *HASHTAG*, *URL*, *ALLCAP*, *CAPFIRST*, *NUMBER*, *EMOJI* and if not any, as *UNK*. The cut-off for what counts as OOV was configurable, although 5 seems to have been a reasonable number.

We used linear interpolation on the transition model for smoothing and add-K smoothing for the emission model and prior models.

From looking at the confusion matrix generated, it seems the greatest source of mistakes were classifications between proper nouns and common nouns, verbs mistaken as proper nouns and between common nouns and verbs. This is most likely due to the treatment of capital letters as part of the word form. This could be explored more, but I am lazy. But the hunch is supported by the fact that if we moved more of the emission symbols into OOV by raising the cut-off, we ended up with more errors in the confusion matrix for those tags. This makes sense as more of the emission symbols were replaced with our word forms.

We tested on different hyper-parameters for the smoothing models and cut-off for OOV (See Table 1. Using the best dev model for $OOVcutoff > 1$ for test resulted in an accuracy of 94.20%. If we were to use the best dev model ($OOVcutof == 1$), the accuracy was only 90.50%.

Question 1.2 Trigram HMM

Question 2 Probabilistic Context Free Grammars