Report

Anonymous ACL submission

1 How did you select the final sets of parameters (N)? What are the values?

We implemented a function to loop over N's value for each smoothing method and selected the N that resulted in the highest accuracy on the development files. For the unsmoothed method, $N = 1$, for add-one smoothing method $N = 4$, and for the interpolation method $N = 3$ provided the best results. We got the accuracy 0.907 for unsmoothed, 0.981 for add-one smoothing, and 0.944 for interpolation methods.

2 Which model performed the best? Discuss the relative performance of the smoothing variants and n-gram settings.

The add-one smoothing method with $N = 4$ resulted in the best accuracy on the development files. Add-one smoothing performed as the best model, and out-performed unsmoothed and interpolation methods. The performance of add-one smoothing decreased for lower N values since there is more information in the longer sequences, which helps to identify the languages better; furthermore, for values greater than $N = 4$, the performance also decreased due to over-fitting and occurrence of more unknown n-grams. Therefore, the $N = 4$ seems a reasonable value that provided the best accuracy for add-one smoothing and among all smoothing methods.

The interpolation method is ranked second based on its performance on the development files. The $N = 3$ was selected as the best value for the interpolation method, and its performance decreased for lower and higher values. The interpolation method assigns lambda values, which are weights for each specific n-gram and for $N = 3$ it takes advantage of uni-grams, bi-grams, and tri-grams. However, for $N = 2$, it only uses uni-grams and bi-grams to estimate the development file's probability. It seems there is some useful information in the tri-grams that helps the interpolation method results in a better performance for $N = 3$ compared to $N = 1$. However, it seems for $N = 4$ the performance decreases which means adding the 4-gram not only helped but also decreased the total performance of the interpolation method. This phenomenon can be explained by paying attention to the training corpus. Since the training corpus is small, there is a high chance that some 4-grams or higher sequences in the development file are not found in the training file. This could be problematic since it causes multiplication of a probability of zero and makes the whole text's probability equal to zero. This was not a significant problem for add-one smoothing since it could prevent the probability of zero by adding one to the counts.

The unsmoothed method had the lowest performance among other methods, and its performance was the best for $N = 1$. Its performance decreased for higher orders of n-grams due to the small size of the training files. This results in very limited sequences in the training file, which produce a probability of zero for some of the sequences in the development file not found in the training file. Therefore, even one unseen sequence can make the probability equal to zero. This cannot be improved by taking advantage of unknown token since these tokens only replace the unseen characters, but we still might get unseen sequences for the higher order of n-grams, and unlike the interpolation method unsmoothed method only utilize one specific n-gram. Therefore, the choice of $N = 1$ seems pretty much reasonable and supports its selection by the unsmoothed method.