Optimization Techniques

1. In stage 1, I order the tokens according to their frequency so that each token has a unique index. Then, each record will be reordered according to the tokens' index, which means those tokens in each record will be listed in ascending order regarding to their frequency.

2. Based on the prefix filtering, those records that share at least one prefix token with each other will be grouped together to make convenient for stage 2. After that, those records that in the same group will be reordered according to their length.
3. In the PPJOIN method, I calculate the ubound which is an upper bound of the overlap between right partitions of x and y with respect to the current token w, whcih is derived from the number of unseen tokens in x and y with the help of the positional information in the index $I_{w.}$

4. In line 34, some prefix tokens only occur in one record, which means no other record will grouped with them based on the token. Thus, filtering these records will save a lot of time.

5. In order to avoid order the key of each result, I order the key pairs in advance(line 116).

6. In line 136, I use reduceBykey because the value of each result is a set. As a result, I don't need to use distinct shuffle function, which save a lot of time. Also, in line 134, I filter the empty result, which reduce a large number of useless results.