Antibody Marathon Solution (a9108tc)

Algorithm

1. The basic idea is to divide and conquer.
    1. We first partite the items into several subsets such that items from different subset is guaranteed to be in different clusters. By simple analysis we know that if two items belongs to the same cluster, there must exist a path between them with each link a similarity smaller than 0.32. So we can just form the subsets by calculating the connectivity of the graph with respect to the links with similarity smaller than 0.32.
    2. Then for each subset, we apply NN-chain algorithm
2. Group the items with the same v_gene,v_all,j_gene, j_all and junc in together before any algorithm is applied. Because if two items share those info, their distance would be 0.001/len(junc), which is close to zero.

Speedup Tricks

1. For many cases, we need to test whether score(i,j) is smaller than a threshold. Sometimes we do not actually need to compute Levenshtein distance to conclude this.
2. We also need to find the items that have similarity less than a threshold to certain given item, say i. By the finding that Sij<Sik+Skj+delta (delta=0.2) for most of the cases, so if we already have Sk*, called the anchor point, then we only need to consider those j with Skj<thres+Ski+delta.

Limitations

1. We only consider links with similarity smaller than THRES_NN (set to 1 in my submission) during the NN-chain algorithm.
2. Each time when we compute the connected subgraph starting with node i, we use i as a anchor point and keep only those Sij<ANCHOR_THRES (set to 1 in my submission). So only those with distance less than ANCHOR_THRES are considered to be in the subgraph.