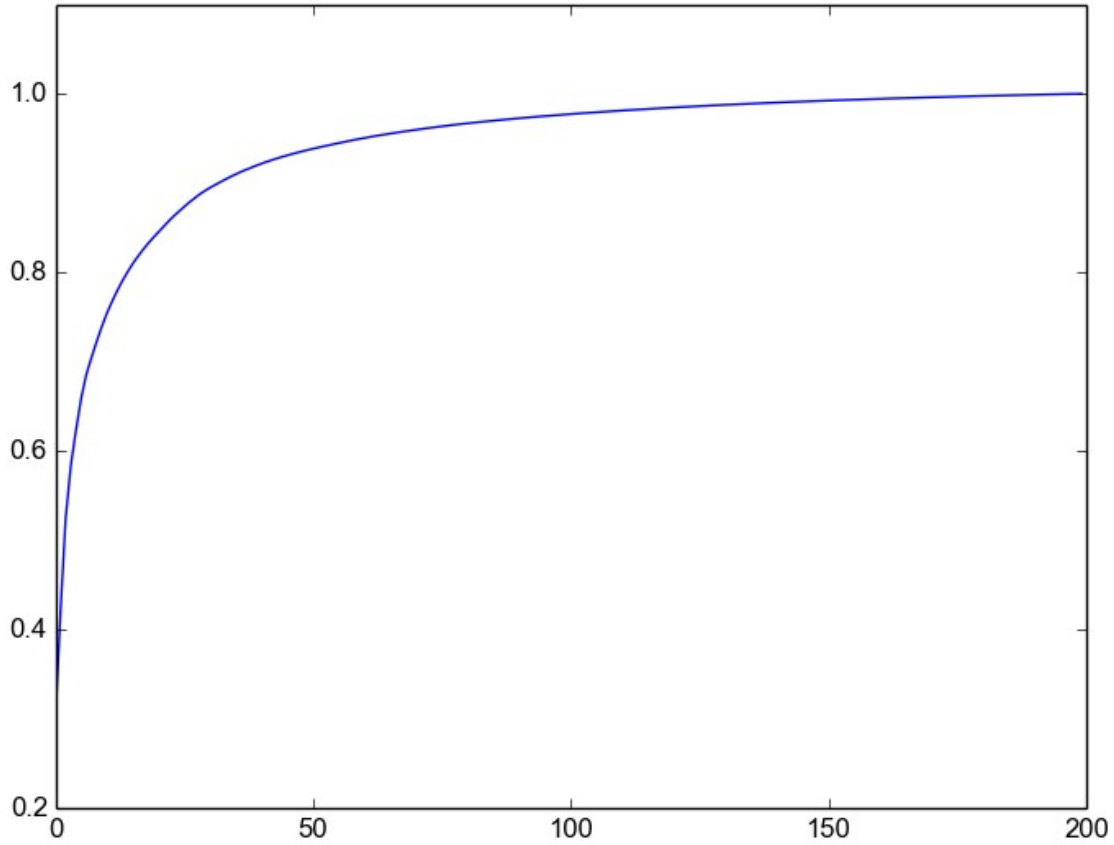
1. Number of sequences = 2500 Number of different commands = 635
Number of co-occurences of rm with ls for user 1 and 2 for the first 5 sequences are:
User 1: 0.0 1.0 3.0 1.0 1.0
User 2: 1.0 0.0 0.0 0.0 0.0
We used the entire command set for training and testing instead of removing the sparse commands.

2. We chose exact mean centering instead of using the sparsity specific mean.
Centered correlation of rm with ls for user 1 and 2 for the first 5 sequences are:
User 1: -0.7176 0.2824 2.2824 0.2824 0.2824
User 2: 0.2824 -0.7176 -0.7176 -0.7176 -0.7176

The contribution ratio was computed as a ratio of contribution of a component to the contribution of top 200 components since computing more components was computationally expensive.



Based on the contribution ratios, we choose the number of principal components that explain around 95% of the variance observed in the top 200 components. The value was found to be 48.

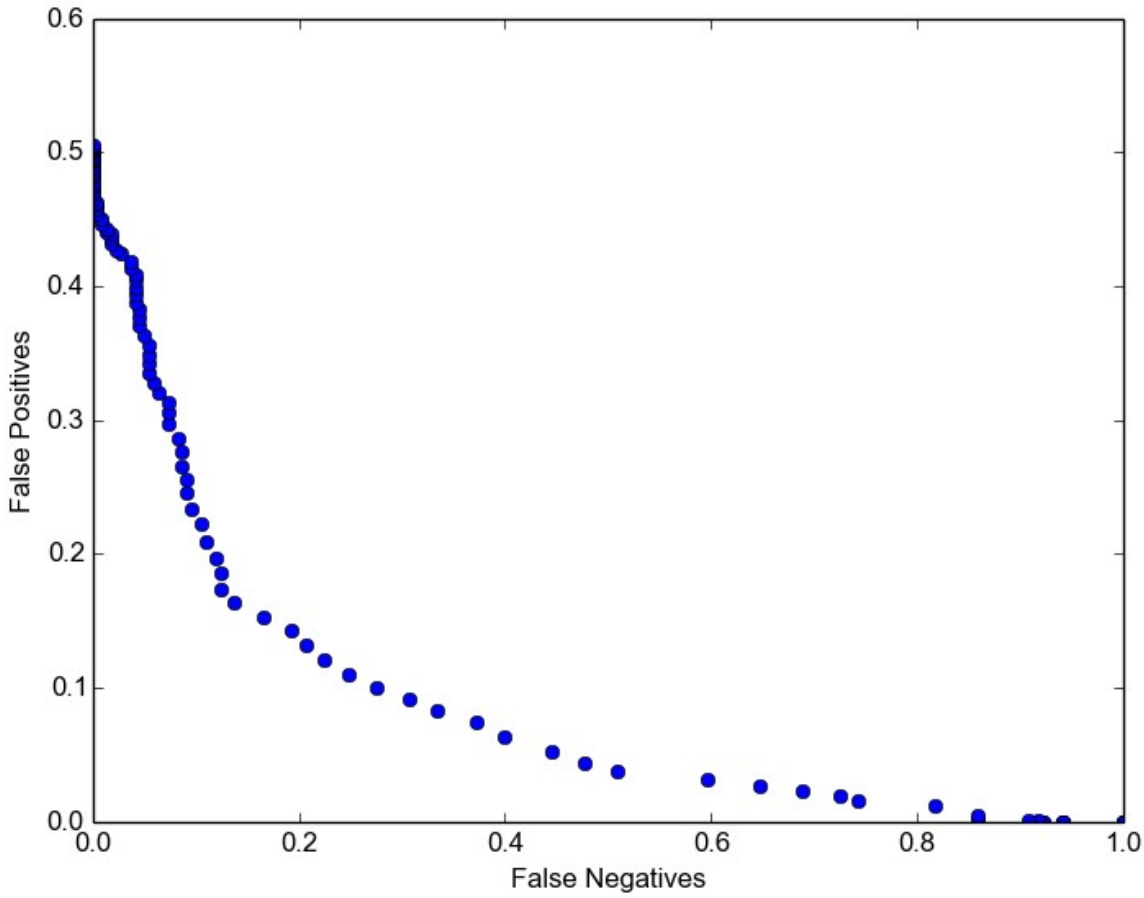
Therefore, the dimension of our reduced space will be N = 48.

3. We chose the layered network model where each layer corresponding to every sequence was stored separately.

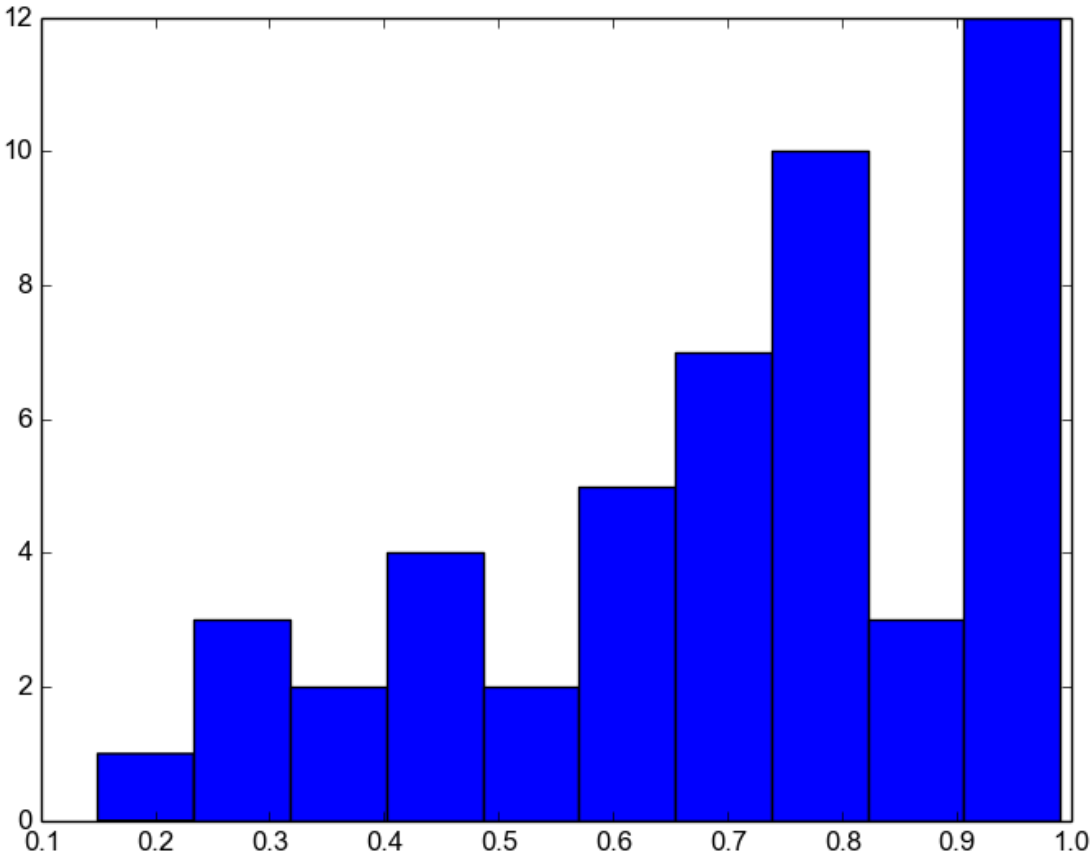
4. Unlike the paper, for finding the similarity of these layered networks, we chose to compare subnetworks of size 2 (in the paper subnetworks of size 3 were used). This was done to avoid the significant computational time required for constructing and comparing these layered networks (the paper reported taking around ~660 minutes for construction and 110 minutes for preparing the lookup tables). By choosing to compare subnetworks of size 2 (which is basically just comparing the edges), we reduced our total construction and lookup time to ~90 minutes. However, this had a detrimental effect on the performance of our system.

5. The first 5 feature vectors for User1 and User2 are attached in user1FV.csv and user2FV.csv

6. We tried several different threshold values (0.1 – 0.5) for constructing the layered networks. We found the best performance for threshold around 0.3, although it didn't offer any significant advantage in terms of the false positives. The observed False positive – False negative observed over 49 users is attached below:



The rest of the results are reported for a similarity cutoff threshold value of 0.34 for which the observed false positive rate over all 49 users was 16.39% and the false negative rate was 13.76%.
However, we observed that the accuracy of the algorithm varied a lot from user to user. A histogram of the accuracy variation over the 49 users is attached:



As we can see, the algorithm performs well for most of the users but has a thin but long tail. After looking at the similarity values over different users, it was observed that the mean similarity varied a lot for different users i.e. some users had very high levels of similarity for all their sequences while others had more variation in their sequences. This combined with using the same threshold for all users would result in low accuracy for some users.

From the results, it seems like having a separate validation set for all users and using this set to determine an appropriate level of threshold might be a good idea to reduce the false positives observed for users with high levels of variations.