Data Mining: Learning from Large Data Sets - Fall Semester 2015

mickell@student.ethz.ch
rabha@student.ethz.ch
rsidhar@student.ethz.ch

October 13, 2015

Approximate near-duplicate search using Locality Sensitive Hashing

Mapper

In our implementation, the signature matrix is split into 8 bands and 2 rows which means that we need a total of 16 min-hash functions. Min-hashing relies on random permutations which are very expensive to generate for 20,001 possible features. For this reason, we approximate permutations by "pseudo-permutations" of the form $\hat{\pi}_i(j) = (a_i \cdot j + b_i) \bmod 20001$ where a_i and b_i are randomly chosen integers. This way, we can compute the signature of each video quite efficiently.

In order to reduce the size of the output, we also hash the rows of the signature of each video. The row vector $\mathbf{r_b}$ of band b is hashed by $h_b(\mathbf{r_b}) = (\mathbf{r_b}^T \cdot \mathbf{a_b} + b_b) \bmod 3367900313$ where a different hash function $h_b(\cdot)$ with different parameters $\mathbf{a_b}$ and b_b is used for every band b. .

The mapper applies the procedure explained above to the features of each input video it receives. Each time, this yields a vector with 8 hashes, one for every band. For each of these hashes, the mapper now outputs one key-value pair, where the key consists of the number of the band and the corresponding hash. The value is the same as the input, that is, the video id and the features.

Reducer

The reducer reads all videos that were assigned the same key by the mapper. Since this is usually a small number of videos, each video can be compared to every other video by computing the Jaccard distance. Only those pairs of videos that have a similarity of at least 90% are then considered duplicates.

What remains to be done is enusure that every pair of duplicates is only printed once. A duplicate pair is passed to a reducer multiple times if and only if multiple bands have the same hash for both videos. Therefore, the reducers recompute the hashes of the two videos any only print the duplicate if the band value in the key corresponds to the lowest band index where both videos have the same hash.

Performance

Our choice to use 8 bands of size 2 ensures fast computation time for the mapper since this corresponds to only 16 hash functions. What is more, the number of videos with equal keys for the reducers is quite

small which makes the reducing step fast. At the same time, the choice of parameters (many bands of small size) makes the probability of false negatives is very low – there are actually none for the training set, we reach a perfect F_1 score of 1.

Collaboration

This was a relatively small project, so decided that each of us solves it individually and we hand in the solution reaching the highest score.