A Comparison of Blocking Methods for Record Linkage

Steorts, Ventura, Sadinle, Fienberg (2014)

Presenter: Christine P. Chai

This paper compares various blocking methods for record linkage, from traditional blocking techniques to locality-sensitive hashing (LSH) related approaches. Traditional blocking methods are generally slow: $O(Bn_{\max}^{d})$ time is required for for B blocks and d databases, where the largest block contains $n_{\max}$ records. For cluster-based blocking, both threshold nearest neighbor clustering (TNN) and K-nearest neighbor clustering (KNN) assign similar records to the same cluster, but the computational complexity is $O(n^2)$. Using canopies (potentially-overlapping sets) is computationally cheap, but the number of canopies is a complicated function determined by the data.

Locality-sensitive hashing (LSH) is a probabilistic algorithm which maps similar items to the same blocks. If two records a, b are close to each other, then the hash value $h(a) = h(b)$ with high probability. If the two records are very different, they have a low probability to hash to the same value. LSH is fast and has high recall (most actual matches included), but it suffers from low precision, i.e. too many false positives.

Transitive locality-sensitive hashing (TLSH) is a variant of LSH, and it maintains transitive closures: If a and b are matches, b and c are matches, then a and c are matches. TLSH starts with shingling each record at letter level and creates a binary matrix M to store the bags of shingles for all n records. Next, we use p minhash functions to map M into an integer-valued matrix M', where each row of M' is a random projection of M. The probability of two columns in M being mapped to the same value equals the Jaccard similarity between the columns. Then the rows of M' are divided into b non-overlapping "bands", so we can apply a real hash function to each band and column.

After the initial setup phase for TLSH, the main steps involve creating a graph and dividing it into connected components. Records are nodes, and edges indicate similarity between records. After creating the connected components, we sub-divide them into "communities", which are dense inside but sparse to the outside. The computational complexity of TLSH is $O(n^2V^{-1})$, dominated by actually building the graph. (V is the number of points in the range of the hash function.)

K-means locality-sensitive hashing (KLSH) also starts with shingling records and implementing random projections. Nevertheless, the similarity between records is measured by the inverse-document-frequency (IDF), and the block assignment is done by K-means. The computational complexity of KLSH is $O(n^{1.5})$, assuming the number of bands increases by $O(\sqrt{n})$. In comparison, traditional non-blocking approaches take $O(n^2)$ time to link two databases.

The blocking results for each algorithm can be measured by the performance on simulated datasets. First, we define a record match as positive and a non-match as negative. Two more terms are defined as below (the higher, the better):[1]

$$\text{Recall} = 1 - FNR = 1 - \frac{FN}{TP+FN} = \frac{TP}{TP+FN} = \frac{\text{\# of true matches found}}{\text{\# of matches generated by the technique}} \tag{1}$$

$$\text{RR (reduction ratio)} = 1 - \frac{\text{\# of matches and non-matches found}}{\text{\# of all records}} \tag{2}$$

For traditional blocking methods, multiple disagreements must exist for a pair to be declared as a non-match. If the method declares non-match based on {first OR last name}, the recall can be less than 40% because this is vulnerable to field errors such as typos. On the other hand, clustering approaches, including TNN and the canopy method, fail to achieve a balance between recall and reduction ratio; that is, the performance is sensitive to the threshold value. Last but not least, TLSH and KLSH perform better in recall than KNN. TLSH has consistently high reduction ratio for shingle number $k \in \{1,2,3,4,5\}$, and KLSH has reduction ratio $\geq 95\%$ as the total block size ≥ 25. However, the performance of TLSH and KLSH depends on tuning parameters, and future work needs to be done to determine which method is preferred in which kind of dataset.

[1] P. Sadosky, A. Shrivastava, M. Price, and R. C. Steorts. Blocking Methods Applied to Casualty Records from the Syrian Conflict. arXiv preprint arXiv,1510.07714, 2015.