CS898D – Data Mining

Homework -2

Chakradhar Reddy Donuri

E949F496

Q1. Explain the steps involved in finding similar documents

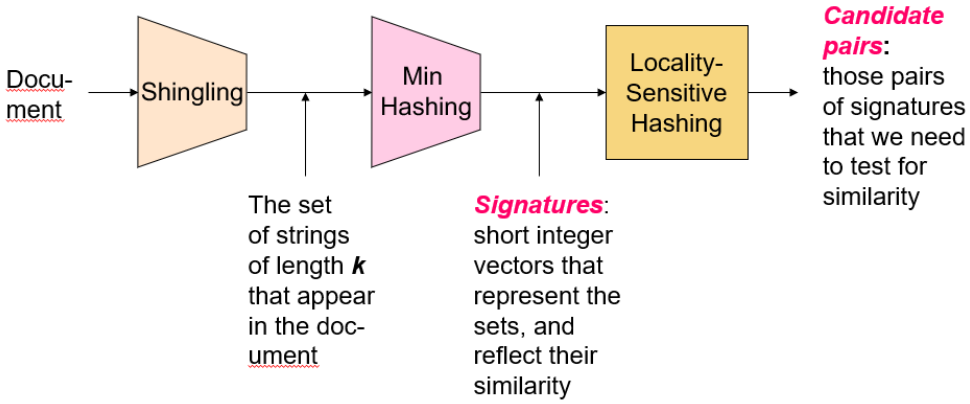
Answer:

3 Essential Steps for Similar Docs

1. ***Shingling:*** Convert documents to sets
2. ***Min-Hashing:*** Convert large sets to short signatures, while preserving similarity
3. ***Locality-Sensitive Hashing:*** Focus on pairs of signatures likely to be from similar documents

1. ***Shingling***:
   - A document consists of many words and we need to consider all the words appearing in the document and also the important words. Apart from this we need to take care of Order in which these words appear so we have a way to encounter this
   - Shingle :
     A *k*-shingle (or *k*-gram) for a document is a sequence of *k* tokens that appears in the doc
     Tokens can be characters, words or something else, depending on the application
     Assume tokens = characters for examples
   - Example: k=2; document $\mathbf{D_1}$ = abcab
     Set of 2-shingles: $\mathbf{S(D_1)}$ = {ab, bc, ca}
     Option**:** Shingles as a bag (multiset), count ab twice: $\mathbf{S'(D_1)}$ **=** {ab, bc, ca, ab}
2. ***Min-Hashing :-***
   - We need to find a hash function $h(\cdot)$ such that:
     if $sim(C_1,C_2)$ is high, then with high prob. $h(C_1) = h(C_2)$
     if $sim(C_1,C_2)$ is low, then with high prob. $h(C_1) \neq h(C_2)$
   - Clearly, the hash function depends on the similarity metric:
     Not all similarity metrics have a suitable hash function
   - There is a suitable hash function for the Jaccard similarity: It is called Min-Hashing
   - Imagine the rows of the boolean matrix permuted under random permutation π
   - Define a "hash" function $h_\pi(C)$ = the index of the first (in the permuted order π) row in which column C has value 1:
     $$h_\pi(C) = min_\pi \pi(C)$$
   - We use several independent hash functions to create a signature of a column
   - We know: $Pr[h_\pi(C_1) = h_\pi(C_2)] = sim(C_1, C_2)$
   - Now generalize to multiple hash functions
   - The *similarity of two signatures* is the fraction of the hash functions in which they agree
3. ***Locality-Sensitive Hashing:***

- We need to find documents with Jaccard similarity at least *s* (for some similarity threshold, e.g., *s*=0.8)
- LSH – General idea: Use a function *f(x,y)* that tells whether *x* and *y* is a *candidate pair:* a pair of elements whose similarity must be evaluated
- For Min-Hash matrices: Hash columns of signature matrix *M* to many buckets and Each pair of documents that hashes into the same bucket is a candidate pair
- Columns *x* and *y* of *M* are a candidate pair if their signatures agree on at least fraction *s* of their rows:
  M (*i, x*) = *M* (*i, y*) for at least frac. *s* values of *i*
- We expect documents *x* and *y* to have the same (Jaccard) similarity as their signatures



Q2. Explain Min-Hashing through an example

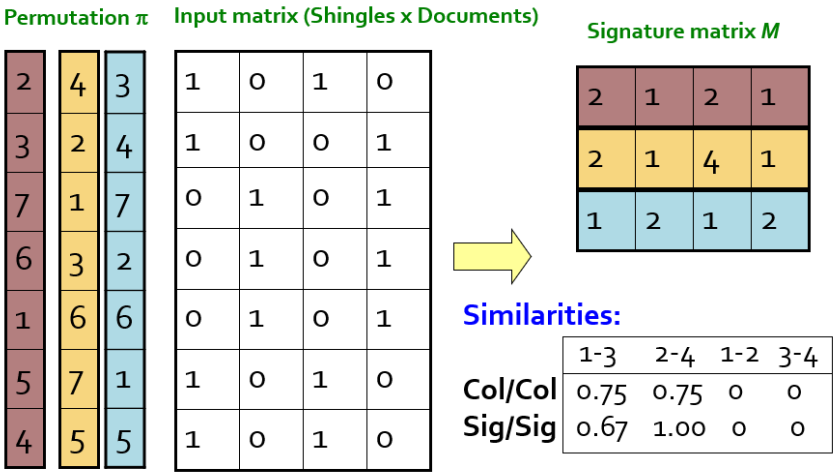
Answer:

Step 1: Randomly permute the items (by permuting the rows of the matrix).

Step 2: Record the first 1 in each column, in which column ***C*** has value **1**

$$h_\pi(C) = \min_\pi \pi(C)$$

Step 3: use the property $\mathbf{Pr}[h_\pi(C_1) = h_\pi(C_2)] = sim(C_1, C_2)$



- Given cols C_1 and C_2, rows may be classified as:

	C_1	C_2
A	1	1
B	1	0
C	0	1
D	0	0

  - a = # rows of type A, etc.

- Note: $sim(C_1, C_2) = a/(a+b+c)$
- Then: $Pr[h(C_1) = h(C_2)] = Sim(C_1, C_2)$
  - Look down the cols C_1 and C_2 until we see a 1
  - If it's a type-*A* row, then $h(C_1) = h(C_2)$
    If a type-*B* or type-*C* row, then not

Implementation:

- **Permuting rows even once is prohibitive**
- **Row hashing!**
  Pick **K = 100** hash functions k_i
  Ordering under k_i gives a random row permutation!
- **One-pass implementation**
  1. For each column ***C*** and hash-func. k_i keep a "slot" for the min-hash value
  2. Initialize all ***sig(C)[i]*** = ∞
     sig(C) = list of P indexes of first rows with 1 in column C
  3. **Scan rows looking for 1s**
     Suppose row ***j*** has 1 in column ***C***
     Then for each k_i :

If $k_i(j) < sig(C)[i]$, then $sig(C)[i] \leftarrow k_i(j)$

Q3. Explain the procedure involved in reducing documents to boolean vectors

Answer:

In order to convert any document to Boolean vectors we need to perform shingling on the documents so, ultimately the output of this process leads to Sets of shingles and then we depict these Sets as Boolean vectors in a matrix as follows

- Consider Boolean (Bits) vectors in reducing a document
- We Encode sets using 0/1 (One dimension per element in the universal set)
- Interpreting the operations set intersection as bitwise AND , set union as bitwise OR
- Example: C_1 = 10111; C_2 = 10011

  Size of intersection = 3; size of union = 4,
  Jaccard similarity (not distance) = 3/4
  Distance: d(C_1,C_2) = 1 – (Jaccard similarity) = ¼

From Sets to Boolean matrices

Documents

Shingles				
	1	1	1	0
	1	1	0	1
	0	1	0	1
	0	0	0	1
	1	0	0	1
	1	1	1	0
	1	0	1	0

- Rows = elements (shingles) and Columns = sets (documents)
- 1 in row *e* and column *s* if and only if *e* is a member of *s*
- Column similarity is the Jaccard similarity of the corresponding sets (rows with value *1)*
- Typical matrix is sparse!
- Each document is a column:

  Example: sim(C_1 ,C_2) = ?
  Size of intersection = 3; size of union = 6,
  Jaccard similarity (not distance) = 3/6
  d(C_1,C_2) = 1 – (Jaccard similarity) = 3/6

Q4 Explain the objective function in K-means clustering. What are the limitations associated with K-means clustering?

Answer:

- In K-means clustering, we seek to partition the observations into a pre-specified number of clusters.
- The idea behind K-means clustering is that a good clustering is one for which the within-cluster variation is as small as possible.
- The within-cluster variation for cluster C_k is a measure WCV(C_k) of the amount by which the observations within a cluster differ from each other. We need to solve

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} \text{WCV}(C_k) \right\}.$$

  Let C1…CK denote sets containing the indices of the observations in each cluster.
  For instance, if the ith observation is in the kth cluster, then i ∈ C_k.
- Using Euclidean distance

$$\text{WCV}(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2,$$

  Where $|C_k|$ denotes the number of observations in the kth cluster.
- Therefore, the optimization problem that defines k-means clustering,

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \right\}$$

- This can be further reduced to

$$\frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = 2 \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2$$

  where $x_{kj} = 1/|C_k| \sum i \in C_k x_{ij}$ is the mean for feature j in cluster C_k.

Limitations :

- Difficult to predict the number of clusters (K-Value)
- Initial seeds have a strong impact on the final results
- The order of the data has an impact on the final results
- Sensitive to scale: rescaling your datasets (normalization or standardization) will completely change results. While this itself is not bad, not realizing that you have to spend extra attention on to scaling your data might be bad.

Q5. Explain the type of linkage in Hierarchical clustering

Answer:

Linkages:

- Complete:

  Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the largest of these dissimilarities.
- Single:

  Minimal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the smallest of these dissimilarities.
- Average:

  Mean inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the average of these dissimilarities.
- Centroid

  Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable inversions.