Applications of LSH

Entity Resolution
Fingerprints
Similar News Articles
Desiderata

◆ Whatever form we use for LSH, we want:

1. The time spent performing the LSH should be linear in the number of objects.
2. The number of candidate pairs should be proportional to the number of truly similar pairs.

◆ Bucketizing guarantees (1).
Entity Resolution

◆ The entity-resolution problem is to examine a collection of records and determine which refer to the same entity.

◆ Entities could be people, events, etc.

◆ Typically, we want to merge records if their values in corresponding fields are similar.
Matching Customer Records

I once took a consulting job solving the following problem:

- Company A agreed to solicit customers for Company B, for a fee.
- They then argued over how many customers.
- Neither recorded exactly which customers were involved.
Company B had about 1 million records of all its customers.

Company A had about 1 million records describing customers, some of whom it had signed up for B.

Records had name, address, and phone, but for various reasons, they could be different for the same person.
Customer Records – (3)

◆ **Step 1**: Design a measure (“score”) of how similar records are:
  ◆ E.g., deduct points for small misspellings (“Jeffrey” vs. “Jeffery”) or same phone with different area code.

◆ **Step 2**: Score all pairs of records; report high scores as matches.
Customer Records – (4)

◆ **Problem**: \((1 \text{ million})^2\) is too many pairs of records to score.

◆ **Solution**: A simple LSH.
  
  - Three hash functions: exact values of name, address, phone.
    - Compare iff records are identical in at least one.
  
  - Misses similar records with a small differences in all three fields.
Aside: Hashing Names, Etc.

◆ How do we hash strings such as names so there is one bucket for each string?

◆ **Possibility**: Sort the strings instead.
  - Used in this story.

◆ **Possibility**: Hash to a few million buckets, and deal with buckets that contain several different strings.

◆ **Note**: these work for minhash signatures/bands as well.
Aside: Validation of Results

We were able to tell what values of the scoring function were reliable in an interesting way.

- Identical records had a creation date difference of 10 days.
- We only looked for records created within 90 days, so bogus matches had a 45-day average.
Validation – (2)

By looking at the pool of matches with a fixed score, we could compute the average time-difference, say $x$, and deduce that fraction $(45-x)/35$ of them were valid matches.

Alas, the lawyers didn’t think the jury would understand.
Validation – Generalized

- Any field not used in the LSH could have been used to validate, provided corresponding values were closer for true matches than false.

- **Example**: if records had a *height* field, we would expect true matches to be close, false matches to have the average difference for random people.
Fingerprint Comparison

- Represent a fingerprint by the set of positions of minutiae.
  - These are features of a fingerprint, e.g., points where two ridges come together or a ridge ends.
LSH for Fingerprints

- Place a grid on a fingerprint.
  - Normalize so identical prints will overlap.
- Set of grid points where minutiae are located represents the fingerprint.
  - Possibly, treat minutiae near a grid boundary as if also present in adjacent grid points.
Discretizing Minutiae

Minutia located here

Maybe pretend it is here also
Applying LSH to Fingerprints

- Make a bit vector for each fingerprint’s set of grid points with minutiae.
- We could minhash the bit vectors to obtain signatures.
- But since there probably aren’t too many grid points, we can work from the bit-vectors directly.
LSH/Fingerprints – (2)

- Pick 1024 (?) sets of 3 (?) grid points, randomly.
- For each set of points, prints with 1 for all three points are candidate pairs.
  - Funny sort of ‘bucketization.’
    - Each set of three points creates one bucket.
    - Prints can be in many buckets.
Example: LSH/Fingerprints

- Suppose typical fingerprints have minutiae in 20% of the grid points.
- Suppose fingerprints from the same finger agree in at least 80% of their points.
- Probability two random fingerprints each have 1 in all three points = $(0.2)^6 = .000064$. 
Example: Continued

- Probability two fingerprints from the same finger each have 1's in three given points: 
  \[ \left( \frac{0.2}{0.8} \right)^3 = 0.004096. \]

- Prob. for at least one of 1024 sets of three points: 
  \[ 1 - (1 - 0.004096)^{1024} = 0.985. \]

- But for random fingerprints: 
  \[ 1 - (1 - 0.000064)^{1024} = 0.063. \]

First image has 1 in a point
Second image of same finger also has 1.
Application: Same News Article

- Recently, the Political Science Dept. asked a team from CS to help them with the problem of identifying duplicate, on-line news articles.

- Problem: the same article, say from the Associated Press, appears on the Web site of many newspapers, but looks quite different.
News Articles – (2)

◆ Each newspaper surrounds the text of the article with:
  ◆ It’s own logo and text.
  ◆ Ads.
  ◆ Perhaps links to other articles.
◆ A newspaper may also “crop” the article (delete parts).
News Articles – (3)

◆ The team came up with its own solution, that included shingling, but not minhashing or LSH.
  ✷ A special way of shingling that appears quite good for this application.
  ✷ LSH substitute: candidates are articles of similar length.
Enter LSH – (1)

- I told them the story of minhashing + LSH.
- They implemented it and found it faster for similarities below 80%.
  - Aside: That’s no surprise. When similarity is high, there are better methods, as we shall see.
Enter LSH – (2)

- Their first attempt at LSH was very inefficient.
- They were unaware of the importance of doing the minhashing row-by-row.
- Since their data was column-by-column, they needed to sort once before minhashing.
New Shingling Technique

- The team observed that news articles have a lot of stop words, while ads do not.
  - “Buy Sudzo” vs. “I recommend that you buy Sudzo for your laundry.”
- They defined a shingle to be a stop word and the next two following words.
Why it Works

- By requiring each shingle to have a stop word, they biased the mapping from documents to shingles so it picked more shingles from the article than from the ads.
- Pages with the same article, but different ads, have higher Jaccard similarity than those with the same ads, different articles.