Most of the slides are from the Mining of Massive Datasets book. These slides have been modified for CS425. The original slides can be accessed at: [www.mindsp.org](http://www.mindsp.org)
Distance Metrics
Distance Measure

- A distance measure $d(x,y)$ must have the following properties:
  1. $d(x,y) \geq 0$
  2. $d(x,y) = 0$ iff $x = y$
  3. $d(x,y) = d(y,x)$
  4. $d(x,y) \leq d(x,z) + d(z,y)$
Euclidean Distance

- Consider two items \( x \) and \( y \) with \( n \) numeric attributes

- Euclidean distance in \( n \)-dimensions:

  \[
  d([x_1, x_2, \ldots, x_n], [y_1, y_2, \ldots, y_n]) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}
  \]

- Useful when you want to penalize larger differences more than smaller ones
L_r-Norm

- **Definition of L_r-norm:**
  \[d([x_1, x_2, \ldots, x_n], [y_1, y_2, \ldots, y_n]) = \left(\sum_{i=1}^{n}|x_i - y_i|^r\right)^{1/r}\]

- **Special cases:**
  - **L_1-norm:** Manhattan distance
    - Useful when you want to penalize differences in a linear way (e.g. a difference of 10 for one attribute is equivalent to difference of 1 for 10 attributes)
  - **L_2-norm:** Euclidean distance
  - **L_\infty-norm:** Maximum distance among all attributes
    - Useful when you want to penalize the largest difference in an attribute
Jaccard Distance

- Given two sets $x$ and $y$:
  \[ d(x, y) = 1 - \frac{|x \cap y|}{|x \cup y|} \]

- Useful for set representations
  - i.e. An element either exists or does not exist

- What if the attributes are weighted?
  - e.g. Term frequency in a document
Cosine Distance

- Consider \(x\) and \(y\) represented as vectors in an \(n\)-dimensional space.

\[
\cos(\theta) = \frac{x \cdot y}{|x||y|}
\]

- The cosine distance is defined as the \(\theta\) value.
  - Or, cosine similarity is defined as \(\cos(\theta)\).

- Only direction of vectors considered, not the magnitudes.
- Useful when we are dealing with vector spaces.
Cosine Distance: Example

\[ \cos(\theta) = \frac{x \cdot y}{||x|| \cdot ||y||} = \frac{0.2 + 0.2 - 0.1}{\sqrt{0.01 + 0.04 + 0.01} \cdot \sqrt{4 + 1 + 1}} = \frac{0.3}{\sqrt{0.36}} = 0.5 \Rightarrow \theta = 60^\circ \]

Note: The distance is independent of vector magnitudes
Edit Distance

- What happens if you search for “Blkent” in Google?
  - “Showing results for Bilkent.”

- **Edit distance** between x and y: Smallest number of insertions, deletions, or mutations needed to go from x to y.

- What is the edit distance between “BILKENT” and “BLANKET”?

  \[
  \begin{align*}
  \text{B L L K E N T} & \quad \text{B I L K E N T} \\
  \text{B L A N K E T} & \quad \text{B L A N K E T}
  \end{align*}
  \]

  \[\text{dist(BILKENT, BLANKET)} = 4\]

- **Efficient dynamic-programming algorithms exist to compute edit distance (CS473)**
Distance Metrics Summary

- Important to choose the right distance metric for your application
  - Set representation?
  - Vector space?
  - Strings?

- Distance metric chosen also affects complexity of algorithms
  - Sometimes more efficient to optimize $L_1$ norm than $L_2$ norm.
  - Computing edit distance for long sequences may be expensive

- Many other distance metrics exist.
Applications of LSH
Entity Resolution
Entity Resolution

- Many records exist for the same person with slight variations
  - Date of birth: “Jan 15, 1957” vs. “1957” vs none
  - Address: Old vs. new, incomplete, typo, etc.
  - Phone number: Cell vs. home vs. work, with or without country code, area code

- Objective: Match the same people in different databases
Locality Sensitive Hashing (LSH)

- Simple implementation of LSH:
  - Hash each field separately
  - If two people hash to the same bucket for any field, add them as a candidate pair
Candidate Pair Evaluation

- Define a scoring metric and evaluate candidate pairs
- Example:
  - Assign a score of 100 for each field. Perfect match gets 100, no match gets 0.
  - Which distance metric for names?
    - Edit distance, but with quadratic penalty
  - How to evaluate phone numbers?
    - Only exact matches allowed, but need to take care of missing area codes.
  - Pick a score threshold empirically and accept the ones above that
    - Depends on the application and importance of false positives vs. negatives
    - Typically need cross validation
Fingerprint Matching
Fingerprint Matching

- Many-to-many matching: Find out all pairs with the same fingerprints
  - Example: You want to find out if the same person appeared in multiple crime scenes

- One-to-many matching: Find out whose fingerprint is on the gun
  - Too expensive to compare even one fingerprint with the whole database
  - Need to use LSH even for one-to-many problem

- Preprocessing:
  - Different sizes, different orientations, different lighting, etc.
  - Need some normalization in preprocessing (not our focus here)
Fingerprint Features

- Minutia: Major features of a fingerprint

- Ridge ending
- Bifurcation
- Short ridge

*Image Source: Wikimedia Commons*
Fingerprint Grid Representation

- Overlay a grid and identify points with minutia
Special Hash Function

- Choose 3 grid points
- If a fingerprint has minutia in all 3 points, add it to the bucket
- Otherwise, ignore the fingerprint.
Locality Sensitive Hashing

- Define 1024 hash functions
  - i.e. Each hash function is defined as 3 grid points

- Add fingerprints to the buckets hash functions

- If multiple fingerprints are in the same bucket, add them as a candidate pair.
Example

- **Assume:**
  - Probability of finding a minutia at a random grid point = 20%
  - If two fingerprints belong to the same finger:
    - Probability of finding a minutia at the same grid point = 80%

- **For two different fingerprints:**
  - Probability that they have minutia at point \((x, y)\)?
    - \(0.2 \times 0.2 = 0.04\)
  - Probability that they hash to the same bucket for a given hash function?
    - \(0.04^3 = 0.000064\)

- **For two fingerprints from the same finger:**
  - Probability that they have minutia at point \((x, y)\)?
    - \(0.2 \times 0.8 = 0.16\)
  - Probability that they hash to the same bucket for a given hash function?
    - \(0.16^3 = 0.004096\)
Example (cont’d)

- For two different fingerprints and 1024 hash functions:
  - Probability that they hash to the same bucket at least once?
    \[ 1 - (1 - 0.04^3)^{1024} = 0.063 \]

- For two fingerprints from the same finger and 1024 hash functions:
  - Probability that they hash to the same bucket at least once?
    \[ 1 - (1 - 0.16^3)^{1024} = 0.985 \]

- False positive rate? 6.3%
- False negative rate? 1.5%
Example (cont’d)

- How to reduce the false positive rate?
- Try: Increase the number grid points from 3 to 6

- For two different fingerprints and 1024 hash functions:
  - Probability that they hash to the same bucket at least once?
    \[ 1 - (1 - 0.04^6)^{1024} = 0.0000042 \]

- For two fingerprints from the same finger and 1024 hash functions:
  - Probability that they hash to the same bucket at least once?
    \[ 1 - (1 - 0.16^6)^{1024} = 0.017 \]

- False negative rate increased to 98.3%!
Example (cont’d)

- Second try: Add another AND function to the original setting
  1. Define 2048 hash functions
     
     *Each hash function is based on 3 grid points as before*
  2. Define two groups each with 1024 hash functions
  3. For each group, apply LSH as before
     
     *Find fingerprints that share a bucket for at least one hash function*
  4. If two fingerprints share at least one bucket in both groups, add them as a candidate pair
Example (cont’d)

- Reminder:
  - Probability that two fingerprints hash to the same bucket at least once for 1024 hash functions:
    - If two different fingerprints: \(1 - (1 - 0.04^3)^{1024} = 0.063\)
    - If from the same finger: \(1 - (1 - 0.16^3)^{1024} = 0.985\)

- With the AND function at the end:
  - Probability that two fingerprints are chosen as candidate pair:
    - If two different fingerprints:
      \[0.063 \times 0.063 = 0.004\]
    - If from the same finger:
      \[0.985 \times 0.985 = 0.97\]

- Reduced false positives to 0.4%, but increased false negatives to 3%

- What if we add another OR function at the end?
Similar News Articles
Similar News Articles

- Typically, news articles come from an agency and distributed to multiple newspapers.

- A newspaper can modify the article a little, shorten it, add its own name, add advertisement, etc.

- How to identify the same news articles?
  - Shingling + Min-Hashing + LSH

- Potential problem: What if ~40% of the page is advertisement? How to distinguish the real article?
  - Special shingling
Shingling for News Articles

- Observation: Articles use stop words (the, a, and, for, …) much more frequently than ads.
- Shingle definition: Two words followed by a stop word.

- Example:
  - Advertisement: “Buy XYZ”
    - No shingles
  - Article: “A spokesperson for the XYZ Corporation revealed today that studies have shown it is good for people to buy XYZ products.”
    - Shingles: “A spokesperson for”, “for the XYZ”, “the XYZ Corporation”, “that studies have”, “have shown it”, “it is good”, “is good for”, “for people to”, “to buy XYZ”.

- The content from the real article represented much more in the shingles.
Identifying Similar News Articles

- High level methodology:
  1. Special shingling for news articles
  2. Min-hashing (as before)
  3. Locality sensitive hashing (as before)