Intelligent Data Management
Edit distance

Michael Debono Mrđen
Bakhodir Ashirmatov

University of Göttingen
Institute of Computer Science

November 28, 2017

Edit distance

- Given two strings a and b the edit distance $d(a, b)$ is the minimum-weight sequence of edit operations transforming a into b

Edit operations

- Insert a symbol - $ab \Rightarrow aXb$
- Delete a symbol - $aXb \Rightarrow ab$
- Replace a symbol - $aXb \Rightarrow aYb$

Edit distance

Examples

- $d('aba','ab') = 1$
- $d('aba','faba') = 1$
- $d('aba','aca') = 1$
- $d('aba','fac') = 3$

Edit distance

Calculation

- Can be computed with dynamic programming in $O(n \cdot m)$

$$dp_{0,0} = 1 - \delta_{00}$$

$$dp_{i,j} = \begin{cases} \min\left(dp_{i-1,j} + 1, i + 1 - \delta_{ij}\right) & \text{if } j = 0 \\ \min\left(dp_{i,j-1} + 1, j + 1 - \delta_{ij}\right) & \text{if } i = 0 \\ \min\left(dp_{i-1,j} + 1, dp_{i,j-1} + 1, dp_{i-1,j-1} + 1 - \delta_{ij}\right) & \text{else} \end{cases}$$

Edit distance

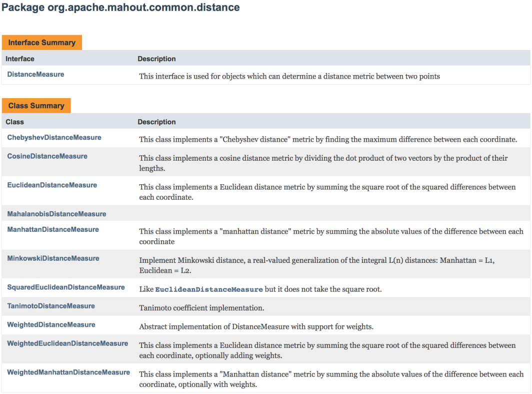
Four distance axioms

- $d(x, x) = 0$ - no edit operations are needed
- $d(x, y) \geq 0$ - number of edit operations is non-negative
- $d(x, y) = d(y, x)$ - edit operations are symmetric
- $d(x, z) \leq d(x, y) + d(y, z)$ - edit operations can be performed consecutively

Edit distance

On Mahout

- Not available
  `https://mahout.apache.org/docs/0.13.1-SNAPSHOT/javadocs/org/apache/mahout/common/distance/package-summary.html`



Package org.apache.mahout.common.distance

Interface Summary

Interface	Description
DistanceMeasure	This interface is used for objects which can determine a distance metric between two points

Class Summary

Class	Description
ChebyshevDistanceMeasure	This class implements a "Chebyshev distance" metric by finding the maximum difference between each coordinate.
CosineDistanceMeasure	This class implements a cosine distance metric by dividing the dot product of two vectors by the product of their lengths.
EuclideanDistanceMeasure	This class implements a Euclidean distance metric by summing the square root of the squared differences between each coordinate.
MahalanobisDistanceMeasure	
ManhattanDistanceMeasure	This class implements a "manhattan distance" metric by summing the absolute values of the difference between each coordinate
MinkowskiDistanceMeasure	Implement Minkowski distance, a real-valued generalization of the integral L(n) distances: Manhattan = L1, Euclidean = L2.
SquaredEuclideanDistanceMeasure	Like `EuclideanDistanceMeasure` but it does not take the square root.
TanimotoDistanceMeasure	Tanimoto coefficient implementation.
WeightedDistanceMeasure	Abstract implementation of DistanceMeasure with support for weights.
WeightedEuclideanDistanceMeasure	This class implements a Euclidean distance metric by summing the square root of the squared differences between each coordinate, optionally adding weights.
WeightedManhattanDistanceMeasure	This class implements a "Manhattan distance" metric by summing the absolute values of the difference between each coordinate, optionally with weights.

Edit distance

Other implementations in other projects

- Java implementation
  https://www.programcreek.com/2013/12/edit-distance-in-java/
- Aamend (Hadoop)
  https://github.com/aamend/hadoop-primitive-clustering
- Fast estimates of Levenshtein Distance on Hadoop
  https://hadoopoopadoop.com/2016/02/12/super-fast-estimates-of-levenshtein-distance/