

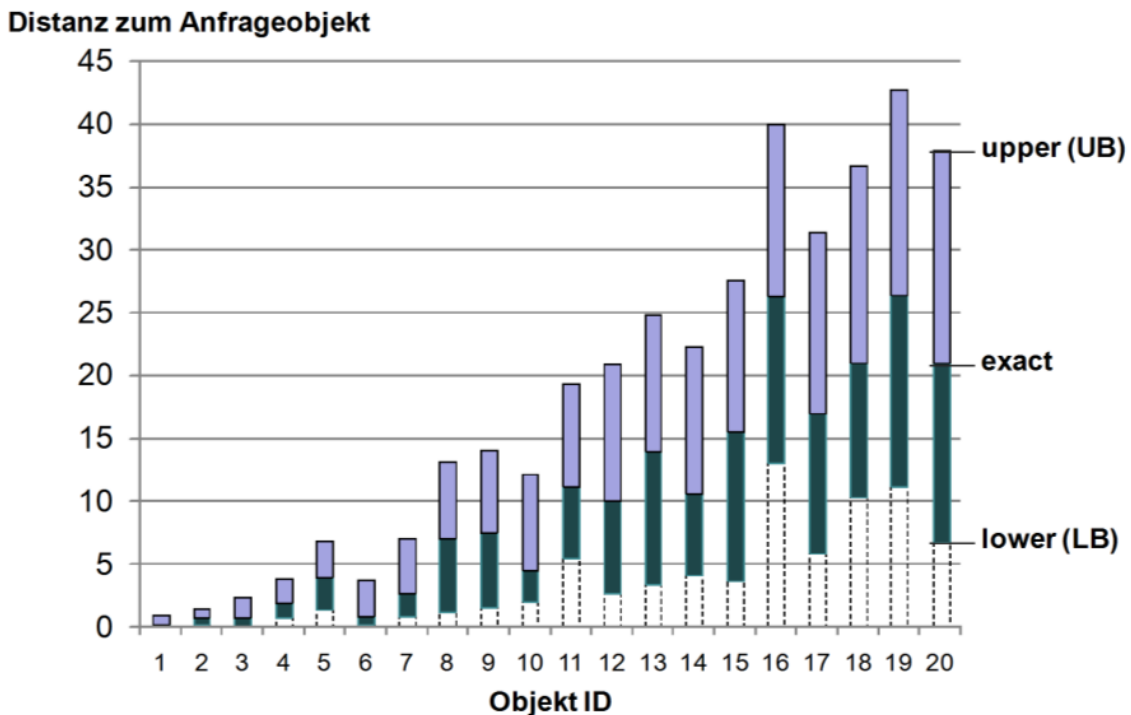
Spatial, Temporal and Multimedia Databases
 WS 2015/16

Übungsblatt 4: Mehrstufige Anfragebearbeitung

Besprechung: 16.11.2015

Aufgabe 4-1 *Mehrstufige Bereichsanfrage*

Gegeben sei eine Menge von Objekten sowie ein Anfrageobjekt Q , auf das eine mehrstufige Bereichsanfrage gestellt wird. Filterdistanzen sowie exakte Distanzen der Objekte zu Q sind im unten abgebildeten Diagramm dargestellt: Objekt-IDs sind auf der x-Achse, zugehörige Distanzwerte auf der y-Achse angetragen. Dabei entspricht der lower-Wert der lower bounding-Distanzabschätzung, der upper-Wert der upper bounding-Distanzabschätzung und der exakt-Wert der genauen Distanz des jeweiligen Objekts zum Anfrageobjekt Q . Die exakten Distanzen sind im Filterschritt nicht verfügbar und können erst nach einem Verfeinerungsschritt verwendet werden.



Ermitteln Sie, welche Objekte bei einer mehrstufigen Bereichsanfrage (RQ-MultiStep, Skript S. 123) mit $\epsilon = 6.0$ zur Resultatmenge gehören. Bestimmen Sie weiterhin, welche Objekte in jedem der folgenden Fälle einem Verfeinerungsschritt zu unterwerfen sind:

- (a) Im Filterschritt ist nur die upper bounding-Distanzabschätzung verfügbar.
- (b) Im Filterschritt ist nur die lower bounding-Distanzabschätzung verfügbar.
- (c) Im Filterschritt sind sowohl die upper als auch lower bounding-Distanzabschätzung verfügbar.

Aufgabe 4-2 Mehrstufige Anfragebearbeitung

In der Grafik (siehe Abbildung) sind sowohl die im Verfeinerungsschritt ermittelten Objektdistanzen als auch die im Filterschritt ermittelten Featurdistancen für eine k -NN-Anfrage mit $k=10$ eingezeichnet. Man ermittle grafisch, wie viele Objekte einem Verfeinerungsschritt unterworfen werden bei folgenden Implementierungsvarianten:

- (a) Auswertung mit Bereichsanfrage (Skript Folie 76)
- (b) Auswertung nach Priorität (Skript Folie 79).

Hinweis: Beide Algorithmen sollen entsprechend auf k -NN ($k \geq 1$) erweitert werden.

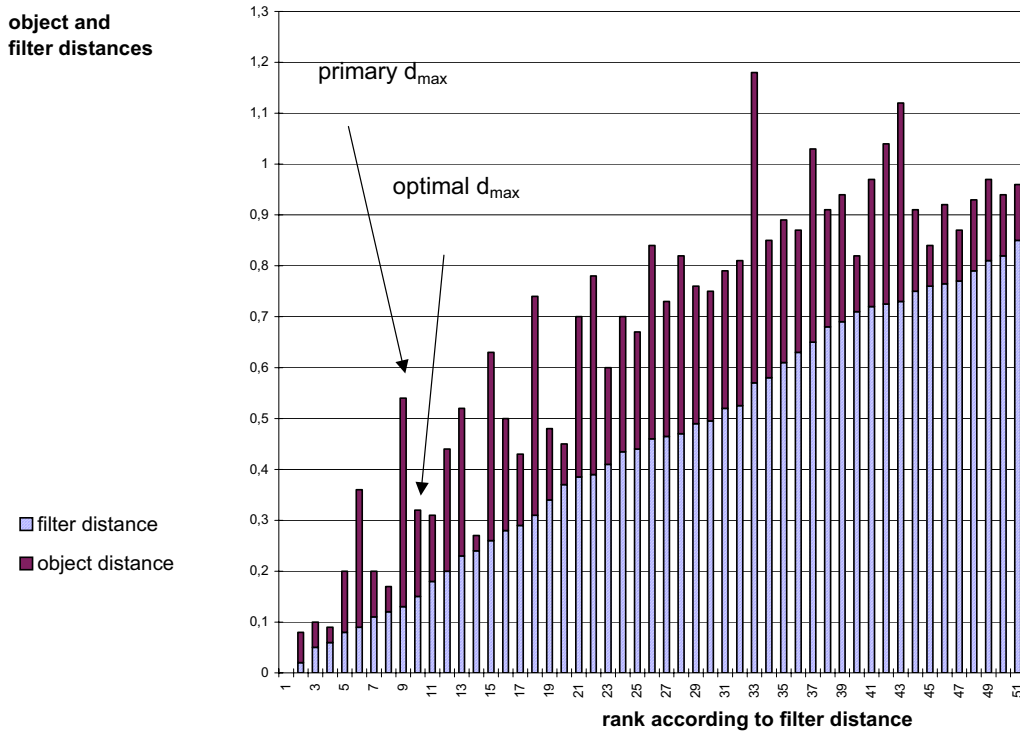


Abbildung 1: Bild zu Aufgabe 3-2

Aufgabe 4-3 *Optimalität der Mehrstufigen Anfragebearbeitung bei Auswertung nach Priorität*

Die *R-Optimalität* einer mehrstufigen k-NN Anfragebearbeitung sei folgendermaßen definiert:

Definition (R-Optimalität): Ein mehrstufiger k-NN-Anfragealgorithmus ist *R-optimal*, wenn im Filterschritt nicht mehr Antwort-Kandidaten erzeugt werden als bezüglich der Filterinformation notwendig.

Wie kann man zeigen, daß die mehrstufigen Anfragebearbeitung bei Auswertung nach Priorität (Algorithmus siehe Skript S. 79) keine *false drops* erzeugt (d.h. keine Ergebnisse verloren gehen) und *R-optimal* ist. (Bem.: Eine Skizze der Beweisidee genügt).