**Informationssysteme SS 2016**
Prof. Dr.-Ing. Sebastian Michel
MSc. Manuel Hoffmann
TU Kaiserslautern, FB Informatik – Lehrgebiet Informationssysteme
http://dbis.informatik.uni-kl.de
**Übungsblatt 1: Ausgabe 19.04.2016, Abgabe 25.04.2016**

# Aufgabe 1: Distanzmaße auf Strings und Dokumenten (1 P.)

a) Geben Sie für die beiden Worte "Rederei" und "Redezeit" die Hamming-Editier-Distanz, Längste-Gemeinsame-Teilsequenz-Distanz und die Levenshtein-Editier-Distanz an. Benutzen Sie für letztere den in der Vorlesung vorgestellten DP-Algorithmus.

b) Wieso sind die folgenden Varianten der Berechnung der Längste-Gemeinsame-Teilsequenz-Distanz nicht sinnvoll?

- $d(x,y) = \max_{s \in S(x,y)} |s|$
- $d(x,y) = min(|x|,|y|) - \max_{s \in S(x,y)} |s|$

c) Gegeben folgende Dokumente:

$d_1 =$ `Frau Neu ist morgen nicht im Büro.`
$d_2 =$ `Frau Neu, ist Prof. Michel morgen im Büro?`

- Betrachten Sie im folgenden die Dokumente ohne Satzzeichen und geben Sie jeweils für $k = 3$ und $k = 4$ die Menge der Shingles der Dokumente an.
- Berechnen Sie jeweils die Ähnlichkeit der Dokumente basierend auf den Mengen der Shingles unter Verwendung des Jaccard-Koeffizienten.
- Berechnen Sie ebenfalls die Ähnlichkeit der Dokumente basierend auf den einzelnen Worten unter Verwendung des Jaccard-Koeffizienten und vergleichen Sie die Resultate.

# Aufgabe 2: Precision und Recall (1 P.)

Stellen Sie sich eine Suchmaschine vor, die eine Anfrage über einem Dokumentenkorpus mit 20 000 Einträgen ausführt und eine geordnete Liste mit 80 Dokumenten zurückgibt. Ein Experte beurteilt die Liste der Ergebnisse und stellt fest, dass die Dokumente auf den folgenden Positionen relevant sind:

$$2, 3, 5, 6, 7, 8, 10, 12, 14, 19,\ \ 21, 23, 29, 30, 37, 42, 46, 48, 50, 55,\ \ 59, 60, 61, 62, 64, 65, 71, 72, 75, 76$$

Weiterhin gibt dieser Experte an, dass der Gesamtkorpus über 400 relevante Dokumente verfügt.

a) Geben Sie für diese Ergebnisliste Precision, Recall, $F_1$-measure, Precision@10 und Precision@20 an.

b) Wie verhalten sich Precision und Recall generell bzgl. Anzahl der Ergebnisse

c) Die Entwickler der Suchmaschine können Ihnen folgende Angebote machen:

- Recall wird erhöht, wenn mehr Zeit für das initiale Sammeln und Verarbeiten der Dokumente aufgewendet wird.
- Precision wird erhöht, wenn für jede einzelne Suchanfrage eine längere Antwortzeit in Kauf genommen wird.

Diskutieren Sie, für welche Einsatzgebiete von Suchmaschinen diese Angebote akzeptabel sind.

# Aufgabe 3: TF*IDF, MMR und LSI (1 P.)

Bei der Analyse klassischer Märchen stellen Sie folgende Verteilung von Termen auf Dokumente $d_1 \dots d_7$ fest:

| **Term** | $d_1$ | $d_2$ | $d_3$ | $d_4$ | $d_5$ | $d_6$ | $d_7$ |
|---|---|---|---|---|---|---|---|
| Vater | 0 | 5 | 0 | 0 | 0 | 1 | 0 |
| Mutter | 2 | 2 | 3 | 2 | 0 | 0 | 3 |
| Königin | 0 | 0 | 0 | 0 | 8 | 1 | 0 |
| Zwerge | 0 | 0 | 0 | 0 | 4 | 0 | 0 |
| Königstochter | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| Wolf | 0 | 0 | 0 | 6 | 0 | 0 | 6 |
| Gold | 2 | 0 | 1 | 0 | 1 | 0 | 0 |
| Haus | 2 | 5 | 1 | 3 | 4 | 1 | 1 |

a) Sortieren Sie die Dokumente für eine Suche nach {Mutter, Haus} nach dem TF*IDF-Modell.

b) Folgende Tabelle enthält die paarweisen Ähnlichkeiten der Dokumente untereinander. Passen Sie die Sortierung aus Teil a) entsprechend diesen Ähnlichkeiten mittels der in der Vorlesung vorgestellten MMR-Methode an, für $\lambda = 0.5$. Berechnen Sie die top-3 Treffer.

| | $d_1$ | $d_2$ | $d_3$ | $d_4$ | $d_5$ | $d_6$ | $d_7$ |
|---|---|---|---|---|---|---|---|
| $d_1$ | 1.0 | 0.16 | 0.14 | 0.18 | 0.16 | 0.14 | 0.16 |
| $d_2$ | 0.16 | 1.0 | 0.16 | 0.40 | 0.18 | 0.18 | 0.16 |
| $d_3$ | 0.14 | 0.16 | 1.0 | 0.16 | 0.15 | 0.16 | 0.12 |
| $d_4$ | 0.18 | 0.40 | 0.16 | 1.0 | 0.17 | 0.16 | 0.17 |
| $d_5$ | 0.16 | 0.18 | 0.15 | 0.17 | 1.0 | 0.18 | 0.15 |
| $d_6$ | 0.14 | 0.18 | 0.16 | 0.16 | 0.18 | 1.0 | 0.13 |
| $d_7$ | 0.16 | 0.16 | 0.12 | 0.17 | 0.15 | 0.13 | 1.0 |

c) Wenden Sie den in der Vorlesung vorgestellten LSI-Ansatz auf die oben angegebene Term-Dokument-Matrix an, für $k = 3$. Verwenden Sie dazu R oder alternativ ein System Ihrer Wahl. Unten finden Sie die ersten Zeilen in R.

- Berechnen Sie die besten Treffer für die Anfrage {Mutter, Haus} und diskutieren Sie die Unterschiede zum Ergebnis aus Teil a).
- Betrachten Sie die Term-Topic-Matrix $U_3$ und diskutieren Sie, in welche Topics LSI die Märchenwelt aufgeteilt ist.

```
n=7  #Anzahl Dokumente
k=3  #Gewuenschter Rang
terme=8  #Anzahl Terme
A<-matrix(c(0,5,0,0,0,1,0,2,2,3,2,0,0,3,0,0,0,0,8,1,0,0,0,0,0,4,0,0,0,0,0,0,
1,1,0,0,0,0,6,0,0,6,2,0,1,0,1,0,0,2,5,1,3,4,1,1), nrow=terme, ncol=n, byrow=TRUE)


#SVD berechnen
A_svd = svd(A)

#und entsprechend dem Rang zuschneiden
u  <- A_svd$u[,1:k]
s  <- diag(A_svd$d)[1:k,1:k]
vt <- t(A_svd$v)[1:k,]

#hier ist die Anfrage im Term-Raum
q1 <- cbind(c(0,1,0,0,0,0,0,1))
....
```