Übung Informationsintegration
String Matching

Aufgabe 1 (Overlap Measure und Jaccard Koeffizient):

Gegeben sind die beiden Stringwerte

$$x = \textbf{'Henri Waternoose'} \text{ und } y = \textbf{'Henry Waternose'}$$

Berechnen Sie das Overlap Measure und den Jaccard Koeffizienten zwischen beiden Werten indem Sie 3-grams zur Tokenbildung verwenden.

Aufgabe 2 (Levenshtein Distanz/Ähnlichkeit):

Gegeben sind die beiden Stringwerte

$$x = \textbf{'Sean'} \text{ und } y = \textbf{'Shawn'}$$

(a) Berechnen Sie die Levenshtein Distanz zwischen beiden Werten mit Hilfe der dynamischen Programmierung.

(b) Leiten Sie die Levenshtein Ähnlichkeit von diesem Ergebnis ab.

Aufgabe 3 (Affine Gap Distance):

Gegeben sind die beiden Stringwerte

$$x = \textbf{'Martin Thomas Doe'} \text{ und } y = \textbf{'Martin T Do'}$$

Berechnen Sie die Affine Gap Distanz zwischen beiden Werten und verwenden Sie dafür die beiden Kosten $w_g = 1$ und $w_s = 0.2$.

Aufgabe 4 (Soundex Code):

Gegeben sind die drei Stringwerte

$$x = \textbf{'depardieu'},\ y = \textbf{'debando'} \text{ und } z = \textbf{'tepadeu'}$$

Berechnen Sie den Soundex Code für x, y und z.

Aufgabe 5 (Extended Jaccard / Generalized Jaccard / Monge-Elkan):

Gegeben sind die beiden Stringwerte

$$x = \textbf{'Tom John Kim'} \text{ und } y = \textbf{'Tim Jon'}$$

(a) Berechnen Sie den Erweiterten Jaccard Koeffizienten und den Generalisierten Jaccard Koeffizienten zwischen beiden Werten. Verwenden Sie den Threshold $\theta = 0.5$.

(b) Berechnen Sie die Monge-Elkan Ähnlichkeit von x zu y.

Verwenden Sie in beiden Fällen die Levenshtein Ähnlichkeit um die Ähnlichkeit zweier Token zu bestimmen.

Aufgabe 6 (TF/IDF):

Gegeben sind die folgenden Firmennamen (bereits in Tokenmengen zerlegt):

$$\begin{aligned}
x_1 &= \{\text{'Insurance', 'Company'}\} \\
x_2 &= \{\text{'A\&B', 'Insurance'}\} \\
x_3 &= \{\text{'BC', 'Company', 'AX'}\} \\
x_4 &= \{\text{'AX', 'Insurance'}\} \\
x_5 &= \{\text{'A\&B', 'Company'}\} \\
x_6 &= \{\text{'XY', 'XY', 'Enterprises'}\}
\end{aligned}$$

(a) Berechnen Sie die *term frequency* und die *inverse document frequency* der einzelnen Token.

(b) Berechnen Sie die Kosinus Ähnlichkeit zwischen x_2 und x_4. Verwenden Sie $TF/IDF(t,d) = tf(t,d) \times idf(t)$ als Wert für das Feature $v_d(t)$.

Aufgabe 7 (Skalierbarkeit):

Gegeben ist ein Stringwert x mit der Tokenmenge $X = \{\text{alpha, delta, zeta}\}$ und eine Menge an Stringwerten $B = \{y_1, \ldots, y_5\}$ mit den Tokenmengen:

Menge B
Y_1 = {alpha, beta, delta, iota}
Y_2 = {gamma, delta}
Y_3 = {alpha, beta, zeta}
Y_4 = {alpha, gamma, iota, zeta}
Y_5 = {alpha, iota}

Zwei Stringwerte werden als Match betrachtet, wenn die Ähnlichkeit ihrer Tokenmengen größer als 0.8 ist. Als Ähnlichkeitsmaß soll der Jaccard Koeffizient verwendet werden.

(a) Erstellen Sie einen invertierten Index und bestimmen Sie mit Hilfe dieses Indexes alle potentiellen Matches von x.

(b) Nutzen Sie das Size Filtering um alle potentiellen Matches von x zu bestimmen.

(c) Nutzen Sie das Prefix Filtering um alle potentiellen Matches von x zu bestimmen. Wählen Sie hierfür den Präfix von X der Länge $|X| - (k - 1)$ (Anmerkung: Beachten Sie das der Jaccard Koeffizient verwendet wird).