Universität Potsdam
Institut für Informatik
Lehrstuhl Maschinelles Lernen

Information Retrieval, Vektorraummodell

Tobias Scheffer
Paul Prasse
Michael Großhans

Information Retrieval

- Konstruktion von Systemen, die die Informationsbedürfnisse der Benutzer befriedigen.
- Repräsentation, Speicherung, Zugriff auf Dokumente.
- Informationsbedürfnis kann, muss aber nicht durch Anfrage ausgedrückt werden.

Information Retrieval
Schlüsselwort-Modell

- Dokument repräsentiert durch Schlüsselwörter.
- Schlüsselwörter unterschiedlich speziell und relevant, unterschiedliche Gewichte.
- $K = \{k_1, \ldots, k_T\}$ sind Index-Terme.
- Dokument $d_J = (w_{1,J}, \ldots, w_{t,J})$.
- $w_{i,J}=0$, wenn k_i nicht in d_J vorkommt, $w_{i,J}=g_i(d_J)$ sonst.

beliebige Funktion; berücksichtigt, ob und wie oft das Wort vorkommt

Information Retrieval
Boolesches Modell

- Dokumente: beschrieben durch Vorkommen von Schlüsselwörtern.
- Suchanfrage sind Boolesche Ausdrücke.
- Ergebnisse der Suchanfrage werden durch Mengenoperationen bestimmt.
- Binäre Entscheidungen, kein Ranking der Ergebnisse.
- Dokument $d_J = (w_{1,J}, \ldots, w_{t,J})$, für Schlüsselwörter.
- $w_{i,J}=0$, wenn k_i nicht in d_J vorkommt, $w_{i,J}=1$ sonst.

Information Retrieval
Ranking oder binäre Entscheidungen?

- Boolsches Modell gibt alle Dokumente zurück, die der Anfrage genügen.
- Keine Reihenfolge.
- Gefühlt „mehr Kontrolle“ für den Nutzer.

- Ranking vs. binäre Entscheidungen:
  - Score-Bewertungen implizieren Ranking.
  - Schlechtes Ranking bedeutet schlechte Performance.

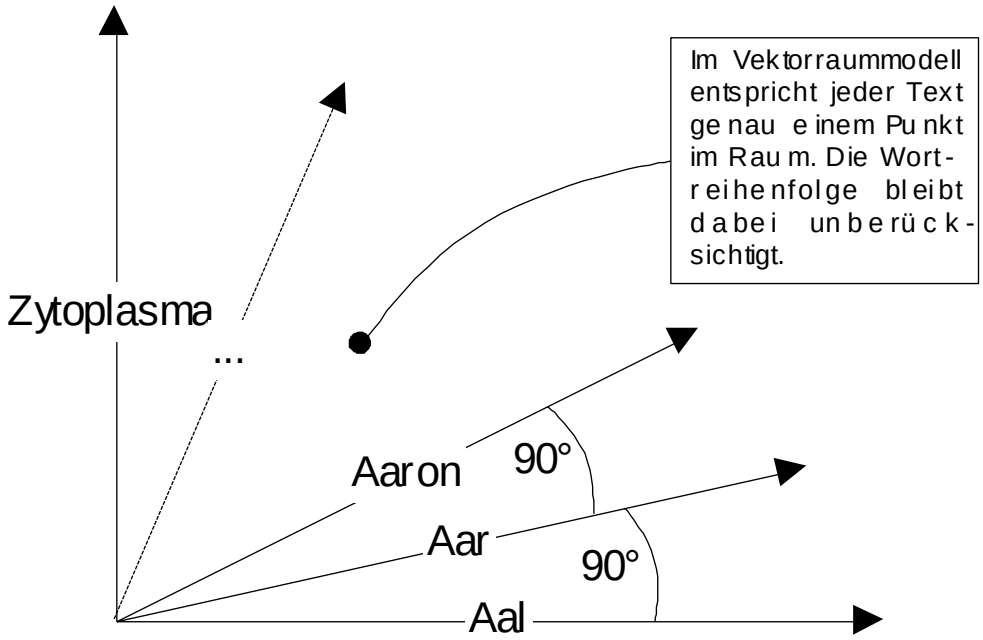
Information Retrieval
Vektorraum-Modell

- Bag-of-Words:
  - Nur die Menge der Wörter wird berücksichtigt.
  - Keine Berücksichtigung der Wortreihenfolge.
- Vektorraummodell:
  - Jeder Text = Punkt in hochdimensionalem Raum.
  - Raum hat eine Dimension für jedes Wort der Sprache.
- Problem: Wörter kommen in unterschiedlichen Formen vor (z.B. Haus vs. Häuser).
  - Stemming.
- Variante: nur Wortstämme berücksichtigen, „Stop-Wörter“ entfernen.

Vektorraum-Modell
Vorverarbeitung

- Stemming: Nur Wortstämme verwenden
  - Auch: bin, bist, sind, … → sein
- Lemmatizing: Wie Stemming, nur unter Zuhilfenahme von Parseinformationen.
- Normalisierung: z.B Bindestriche zwischen Wörtern entfernen.
- Auch Synonymauflösung: Auto, Wagen, …
- Stopwords: der, die, das, von, …

Vektorraum-Modell



- Text wird repräsentiert durch Punkt im hochdimensionalen Raum,
- Wortreihenfolge bleibt unberücksichtigt,
  - „Fußball ist toll“ und „Toll ist Fußball“ gleich.
- Variante: Wortstammbildung, „inverse document frequency“.

Vektorraum-Modell
TFIDF-Repräsentation

- Termfrequenz eines Wortes in einem Text = # Vorkommen des Wortes im Text.
- Problem 1: Einige Wörter sind weniger relevant
  - Z.B. und, oder, nicht, wenn, …
- Lösung: Inverse Dokumentenfrequenz

$$IDF(wort_i) = \log \frac{\#\text{Dokumente}}{\#\text{Dokumente , in denen Wort}_i \text{ vorkommt}}$$

Vektorraum-Modell
TFIDF

- Problem 2: Lange Texte haben lange Vektoren (hohe Vektornorm), führt zu Verzerrungen beim Ähnlichkeitsmaß.
- Lösung: Normieren.

- Repräsentation eines Textes:

$$TFIDF(Text) = \frac{1}{norm} \begin{pmatrix} TF(Wort_1) \cdot IDF(Wort_1) \\ \vdots \\ TF(Wort_n) \cdot IDF(Wort_n) \end{pmatrix}$$

Alle Vektoren haben die Länge 1.

Vektorraum-Modell
TFIDF

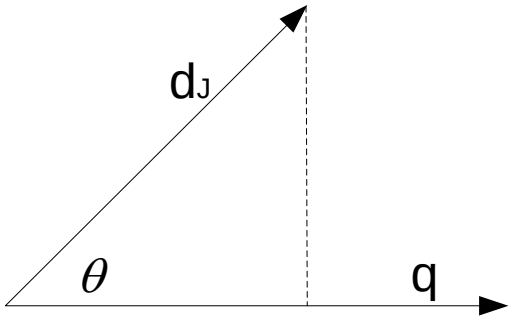
- Alternative Definitionen für Termfrequenz, Dokumentfrequenz und Normalisierung

Term frequency		Document frequency		Normalization	
n (natural)	$\mathrm{tf}_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1+\log(\mathrm{tf}_{t,d})$	t (idf)	$\log \frac{N}{\mathrm{df}_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2+w_2^2+\ldots+w_M^2}}$
a (augmented)	$0.5+\frac{0.5\times \mathrm{tf}_{t,d}}{\max_t(\mathrm{tf}_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N-\mathrm{df}_t}{\mathrm{df}_t}\}$		
b (boolean)	$\begin{cases} 1 & \text{if } \mathrm{tf}_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$				

- Auszug aus Manning, et.al.: Introduction to Information Retrieval: http://nlp.stanford.edu/IR-book/

Vektorraum-Modell
Ähnlichkeitsmaß

- Ähnlichkeit zwischen Text d_J und und Anfrage q: Cosinus des Winkels zwischen den Vektoren.



- Ähnlichkeit: $sim(d_j, q) = \cos(\theta) = \frac{d_j \cdot q}{|d_j| \cdot |q|}$
- Zwischen 0 und 1.

Probabilistisches Modell

- Binary independence retrieval (BIR) model.
- Dokument $d_J = (w_{1,J}, \ldots, w_{t,J})$, für Schlüsselwörter.
- $w_{i,J}=0$, wenn k_i nicht in d_J vorkommt, $w_{i,J}=1$ sonst.
- R ist die Menge der relevanten Dokumente.
- Gesucht: Schätzer für $P(R \mid d_J)$: P(Dokument ist relevant).
- Ähnlichkeit: Odds-Ratio

$$sim(d_j, q) = \frac{P(R \mid d_j)}{P(\bar{R} \mid d_j)}$$

Probabilistisches Modell

- Bayes‘ Regel: $sim(d_j, q) = \frac{P(R \mid d_j)}{P(\bar{R} \mid d_j)}$

$$= \frac{P(d_j \mid R)P(R)}{P(d_j \mid \bar{R})P(\bar{R})}$$

- $P(R)$ und $P(\bar{R})$ ist konstant
  - für alle Dokumente gleich

$$sim(d_j, q) \sim \frac{P(d_j \mid R)}{P(d_j \mid \bar{R})}$$

- Annahme: Die Terme des Dokuments sind unabhängig:

$$P(d_j \mid R) = \left(\prod_{i:w_{ij}=1} P(k_i \mid R)\right)\left(\prod_{i:w_{ij}=0} P(\bar{k}_i \mid R)\right)$$

Schlüsselwort ist enthalten

Schlüsselwort ist nicht enthalten

Probabilistisches Modell

- Dann folgt für sim:
- $$sim(d_j, q) \sim \frac{\left(\prod_{i:w_{ij}=1} P(k_i \mid R)\right)\left(\prod_{i:w_{ij}=0} P(\bar{k}_i \mid R)\right)}{\left(\prod_{i:w_{ij}=1} P(k_i \mid \bar{R})\right)\left(\prod_{i:w_{ij}=0} P(\bar{k}_i \mid \bar{R})\right)}$$
- $P(k_i \mid R)$: Wahrscheinlichkeit für Anfrageterm k_i in relevanten Texten.
- $P(\bar{k}_i \mid R)$: Wahrscheinlichkeit dafür, dass Anfrageterm k_i in relevanten Texten nicht auftritt.

Probabilistisches Modell

- Annahme: Wörter, die in der Anfrage q nicht auftauchen, haben gleiche Wahrscheinlichkeit in relevanten wie nicht relevanten Dokumenten vorzukommen.

- Logarithmiert und leicht umgeformt:

$$sim(d_j, q) \sim \sum_{i=1}^{t} w_{iq} \cdot w_{ij} \cdot \left(\log \frac{P(k_i \mid R)}{1 - P(k_i \mid R)} + \log \frac{1 - P(k_i \mid \bar{R})}{P(k_i \mid \bar{R})} \right)$$

Probabilistisches Modell

- Problem: $P(k_i \mid R)$ ist nicht bekannt.
- Muss irgendwie von Hand eingestellt oder aus Daten gelernt werden.

Zusammenfassung

- Arten Texte darzustellen:
  - Schlüsselwortmodell,
  - Bag-of-Words,
  - TFIDF-Repräsentation.

- Arten die Relevanz eines Dokuments zu einer Anfrage zu bestimmen:
  - Testen auf größte Übereinstimmung,
  - Kosinus-Ähnlichkeit,
  - Probabilistisches Modell.

Fragen?