Information Retrieval und Question Answering

Kai Kugler

19. November 2009

Auffinden von relevantem Wissen

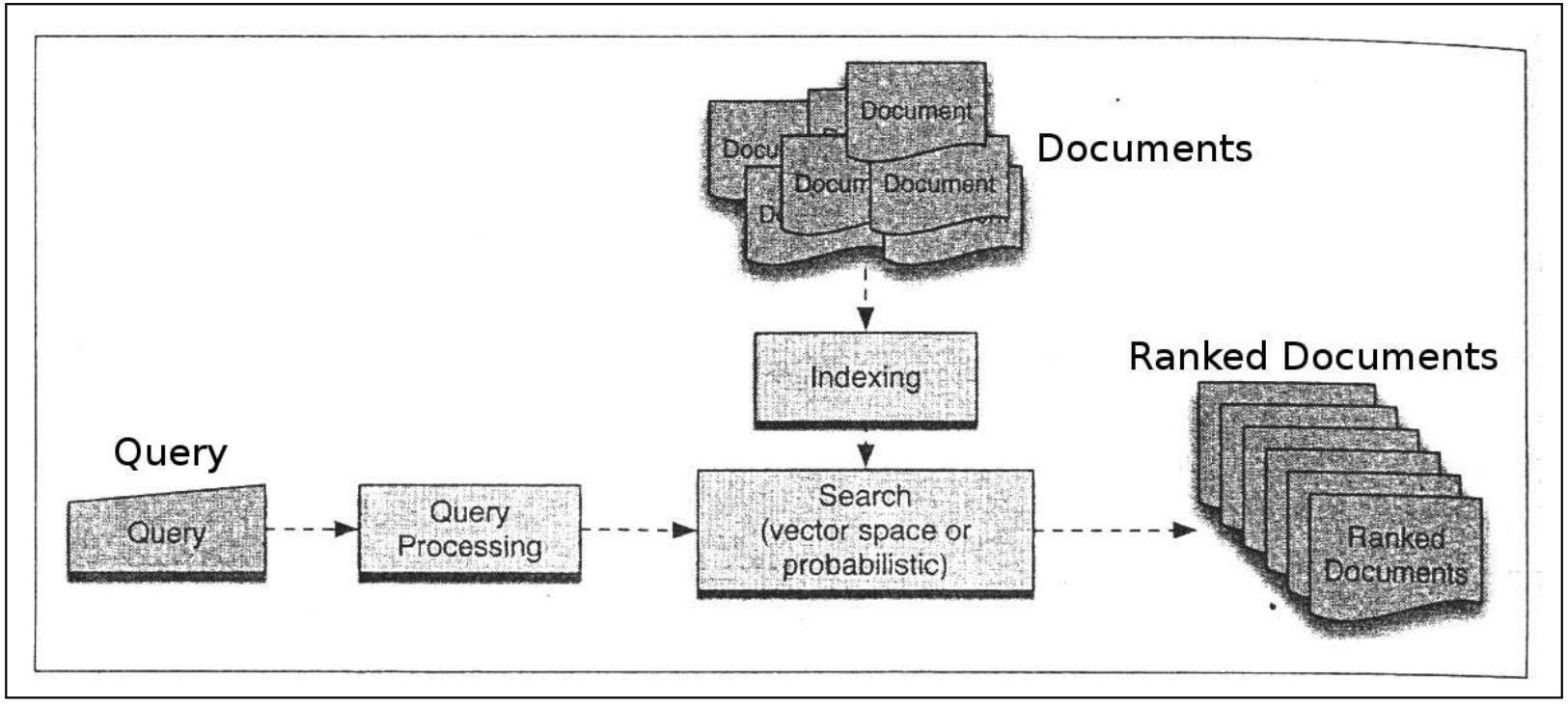
Die Relevanz der aufzufindenden Information ist abhängig vom ...

- aktuellen Wissen des Benutzers
- dem aktuellen Problem des Benutzers
- der subjektiven Erwartung des Benutzers

Die Anfrage ist ...

- nicht präzise
- nicht formal
- vage/unscharf

Architektur eines ad hoc IR-Systems



Repräsentation der Dokumente

Dokumente werden als Vektor ihrer Merkmale dargestellt.
Beispiel (einfache Termhäufigkeit):
Zeitungsartikel mit den Termen
(Naumann, Kulturbeauftragte, parlamentarisch, Staatssekretär)
und den Häufigkeiten (11,14,12,9) ein anderer mit den
Häufigkeiten (8,0,0,0).
Vektoren $\vec{d}_1 = (11, 14, 12, 9)$ und $\vec{d}_2 = (8, 0, 0, 0)$
Matrix:

$$A = \begin{pmatrix} 11 & 8 \\ 14 & 0 \\ 12 & 0 \\ 9 & 0 \end{pmatrix}$$

Repräsentation der Dokumente

Allgemein:

$$\vec{d}_j = (w_{1,j}, \cdots, w_{n,j})$$

(Dokument j (aus einem Textkorpus mit n Termen) als Vektor seiner Termgewichte (w))

Ebenso: Anfrage als Vektor der Terme

$$\vec{q} = (w_{1,q}, \cdots, w_{n,q})$$

Repräsentation der Dokumente

Allgemein:

$$\vec{d}_j = (w_{1,j}, \cdots, w_{n,j})$$

(Dokument j (aus einem Textkorpus mit n Termen) als Vektor seiner Termgewichte (w))
Ebenso: Anfrage als Vektor der Terme

$$\vec{q} = (w_{1,q}, \cdots, w_{n,q})$$

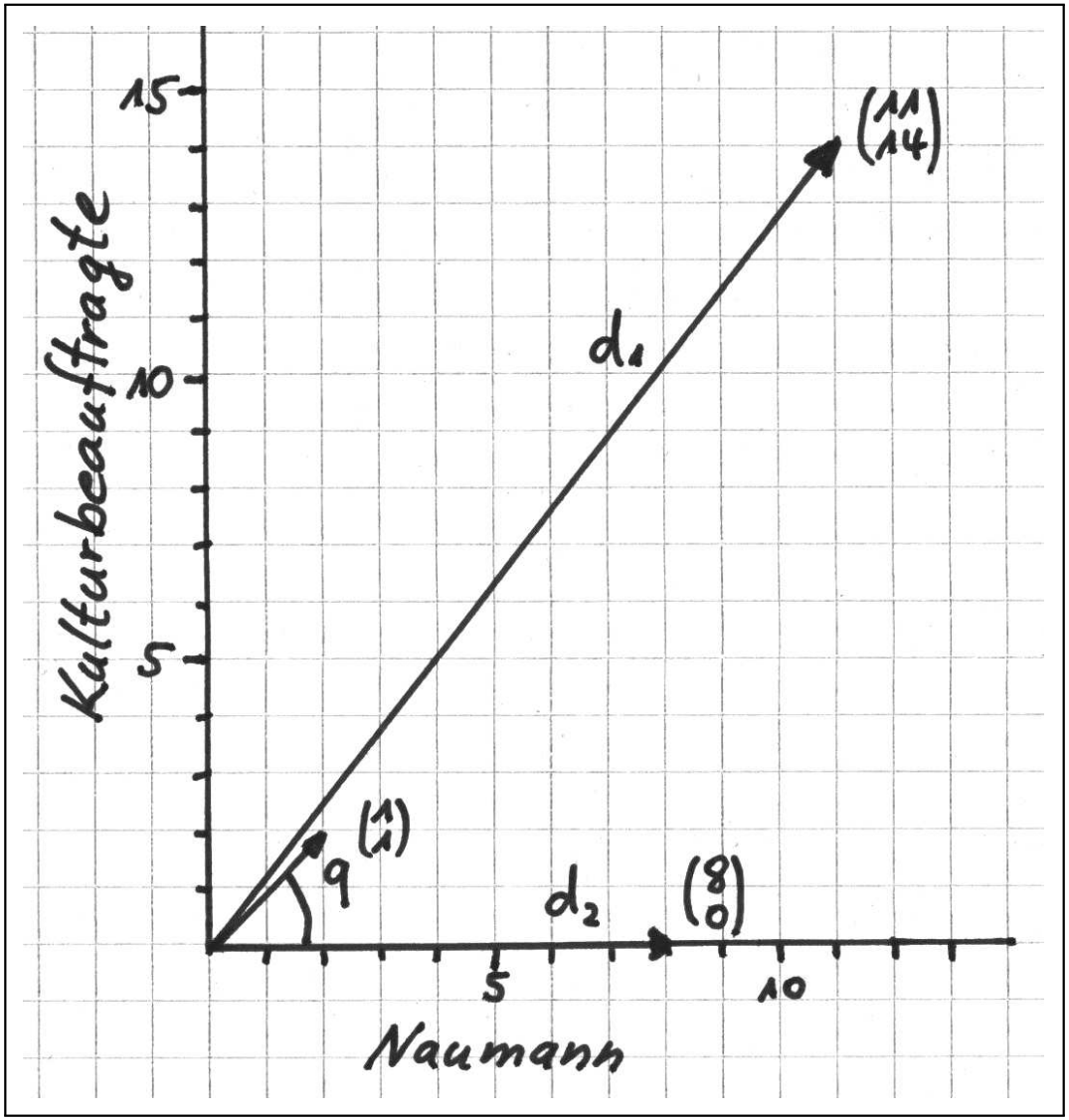
Vergleich der Vektoren: Cosinus

Cosinus des jeweiligen Winkels zwischen $\vec{q}$ und $\vec{d}_1$ bzw. $\vec{d}_2$:

$$sim(\vec{q}, \vec{d}_j) = \frac{\sum_{i=1}^{N} w_{i,q} \times w_{i,j}}{\sqrt{\sum_{i=1}^{N} w_{i,q}^2} \times \sqrt{\sum_{i=1}^{N} w_{i,j}^2}}$$

Hierbei gilt: $sim(\vec{q}, \vec{d}_j) \in [0, 1]$ (bei identischen Vektoren 0, bei orthogonalen 1)

Vergleich der Vektoren: Cosinus



Termgewichtung

Tf-idf (term frequency - inverse document frequency) ist das Produkt aus Termhäufigkeit (tf) und inverser Dokumentenhäufigkeit (idf):

$$w_{i,j} = tf_{i,j} \cdot idf_i = \frac{freq_{i,j}}{max_l(freq_{l,j})} \cdot \log \frac{N}{n_i}$$

($w_{i,j}$ Gewicht des Terms i im Dokument j, $freq_{i,j}$ Häufigkeit des Terms i in Dokument j, $max_l(freq_{l,j})$ Maximalhäufigkeit aller Terme im Dokument, N Anzahl der Dokumente im Korpus, n_i Anzahl der Dokumente, in denen Term i vorkommt)

Beispiel: Termgewichtung

term	**tf-idf**
Kulturbeauftragte	0.0412
parlamentarisch	0.0295
Naumann	0.0284
Staatssekretär	0.0233
Bundeskulturbeauftragter	0.0186
Kulturpolitik	0.0147

(Das Dokument kann z.B. entsprechend verschlagwortet werden)

Termgewichtung

Diese Termgewichtung kann beim Vergleich (Cosinus) der Vektoren benutzt werden:

$$sim(\vec{q}, \vec{d}) = \frac{\sum_{w \in q,d} tf_{w,q}\ tf_{w,d}\ (idf_w)^2}{\sqrt{\sum_{q_i \in q} (tf_{q_i,q} idf_{q_i})^2} \times \sqrt{\sum_{d_i \in d} (tf_{d_i,d} idf_{d_i})^2}}$$

Termauswahl

- Stemming

word	stem
auffallen	auffall
auffallend	auffall
auffallenden	auffall
auffällig	auffall
katz	katz
katze	katz
katzen	katz

- Stoppwortliste
  z.B.: auch, auf, aus, bis, . . .
  Problem Phrasensuche: *to be or not to be* $\longrightarrow$ *not*

Probleme

- Synonyme $\longrightarrow$ nicht alle relevanten Dokumente
- Polyseme $\longrightarrow$ auch irrelevante Dokumente
- Homonyme $\longrightarrow$ auch irrelevante Dokumente

Evaluation von IR-Systemen

Relevante vs. irrelevante Dokumente.
Berechnung precision und recall:

$$\begin{aligned} \mathit{precision} &= \frac{|\mathit{retrieved} \cap \mathit{relevant}|}{|\mathit{retrieved}|} \\ \mathit{recall} &= \frac{|\mathit{retrieved} \cap \mathit{relevant}|}{|\mathit{relevant}|} \end{aligned} \tag{1}$$

Beispiel: Berechnung precision und recall

Das Korpus umfasst 6 Dokumente, davon sind 4 relevant (*relevant*) und 2 irrelevant.
Das IR-System liefert für eine bestimmte Anfrage 3 Dokumente (*retrieved*), von denen 2 ($|\mathit{retrieved} \cap \mathit{relevant}|$) korrekt als relevant und eines fälschlicher Weise als relevant eingestuft wurde.
Somit ergeben sich eine precision von $\frac{2}{3} = 0.\bar{6}$ und ein recall von $\frac{2}{4} = 0.5$.

Verbesserung der Ergebnisse

- Bewertung durch den Benutzer
  Sei $\vec{q_i}$ der ursprüngliche Anfragevektor, $\vec{r}$ (relevant) und $\vec{s}$ (irrelevant) Dokumente, R die Anzahl der relevanten Dokumente und S die der irrelevanten, $\beta, \gamma \in [0, 1]$.

$$\vec{q}_{i+1} = \vec{q_i} + \frac{\beta}{R} \sum_{j=1}^{R} \vec{r}_j - \frac{\gamma}{S} \sum_{k=1}^{S} \vec{s}_k$$

- Erweiterung des Anfragevektors durch Terme aus
  - relevanten Dokumenten
  - dem Umfeld der Anfrageterme (passage retrieval)
  - einem Thesaurus (Synonyme, Hyponyme, Hyperonyme...)

Verbesserung der Ergebnisse

- Bewertung durch den Benutzer
  Sei $\vec{q_i}$ der ursprüngliche Anfragevektor, $\vec{r}$ (relevant) und $\vec{s}$ (irrelevant) Dokumente, R die Anzahl der relevanten Dokumente und S die der irrelevanten, $\beta, \gamma \in [0, 1]$.

$$\vec{q}_{i+1} = \vec{q_i} + \frac{\beta}{R} \sum_{j=1}^{R} \vec{r}_j - \frac{\gamma}{S} \sum_{k=1}^{S} \vec{s}_k$$

- Erweiterung des Anfragevektors durch Terme aus
  - relevanten Dokumenten
  - dem Umfeld der Anfrageterme (passage retrieval)
  - einem Thesaurus (Synonyme, Hyponyme, Hyperonyme...)

WordNet

retrieval	(computer science) the operation of accessing information from the computer's memory
retrieval	the cognitive operation of accessing information in memory
recovery	the act of regaining or saving something lost (or in danger of becoming lost)

Hyperonyme für retrieval (computer science):

- computer operation
- operation
- data processing
- processing

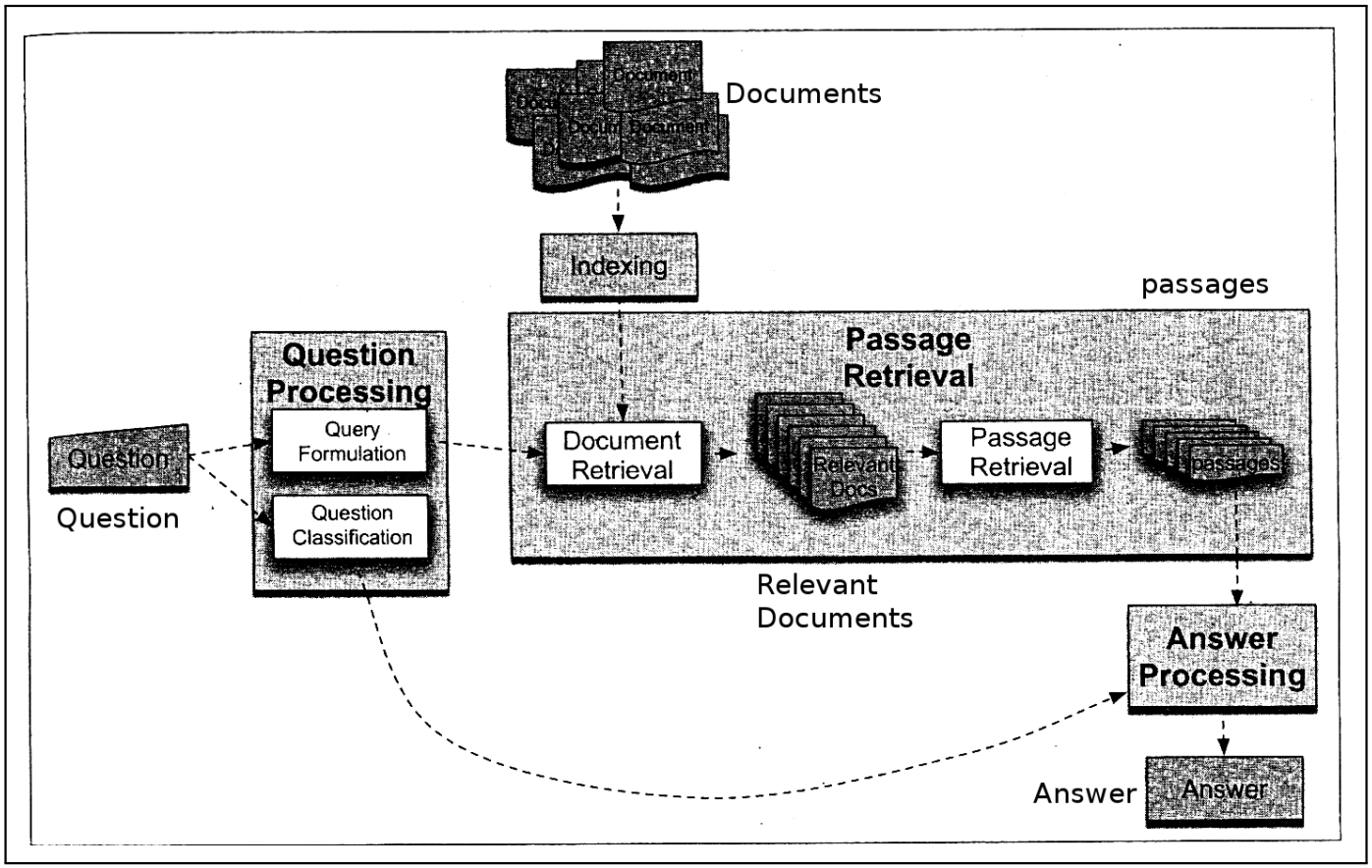
Factiod Question Answering

Fragen, die einen einfachen Fakt als Antwort haben.

- What currency is used in China? $\rightarrow$ the yuan
- Where is the Louvre located? $\rightarrow$ in Paris, France

Ziel: Antworten mit Hilfe von IR aus Dokumenten ermitteln

Drei Phasen des QA



question processing → passage retrieval → answer processing

Question processing

(Vorverarbeitung der Anfrage)

- Query formulation
  Erweiterung der Anfrage (Thesaurus) oder regelbasiertes Umformulieren:
  Where is A $\rightarrow$ *A is located in*
  where is the valley of kings? $\rightarrow$ *the valley of kings is located in*

Question processing

- Question Classification
  Klassifizieren der Frage (Taxonomie), z.B. regelbasiert:

$$who\ \{is|was|are|were\}\ PERSON$$

  Hinweise auf den Typ:

  - Fragewort z.B. *Wer*... $\rightarrow$ Person
  - Schlüsselwörter z.B.
    Welche Stadt... $\rightarrow$ Ort
    ...*Geburtstag*... $\rightarrow$ Datum
  - Eigennamen

Passage Retrieval

Nur Textpassagen der relevanten Dokumente beachten, die Informationen entsprechend des Fragetyps beinhalten.
Diese Textpassagen werden rangiert, die irrelevanten verworfen.
Rang nach:

- Anzahl der Eigennamen vom richtigen Typ
- Anzahl übereinstimmender Schlüsselwörter
- längste übereinstimmende Sequenz von Schlüsselwörtern
- Rang des Dokuments
- Wortabstand der Schlüsselwörter
- N-gram-Überlappung

Answer processing

Extraktion der Antwort aus den relevanten Textpassagen

- nach Antworttyp
  - ist der Antworttyp `HUMAN`, muß nur noch nach entsprechenden Entitäten gesucht werden
  - Auffinden in den Textpassagen z.B. mit named entity taggern
- nach Antwortmuster

Question	**Pattern**	**Answer**
What is a <caldera>?	<QP>, a <AP>	the Long Valley <caldera>, a <volcanic crater> 19 miles long

(Verleich mit regulären Ausdrücken)

Antwortmuster

Manuell erstellt oder mit Lernalgorithmen, z.B. für die Realation `PERSON-NAME/YEAR-OF-BIRTH`:

- Startend mit der Relation (`person-name` → `year-of-birth`) und gegebenen Termpaaren (`gandhi:1869`, `mozart:1756`)
- Dokumentensuche anhand der Paare (IR)
- Extraktion der Sätze, die dieses Paar beinhalten
- Erzeugen eines entsprechenden Musters
- Testen des Musters anhand des Termpaares (liefert das Muster für `mozart` tatsächlich `1756`?)

Für die Relation (`person-name` → `year-of-birth`) und den Antworttyp `YEAR-OF-BIRTH` würde man z.B. folgende Muster erwarten:

- `<NAME> (<BD> - <YD>)`
- `<NAME> was born on <BD>`