



ZBIW Zertifikatskurs Data Librarian

Suchmaschinentechnologie, Information Retrieval

Björn Engelmann, Technische Hochschule Köln, Cologne, Germany
Übernommen von Prof. Schaer

Technology
Arts Sciences
TH Köln

Was ist eigentlich Retrieval?

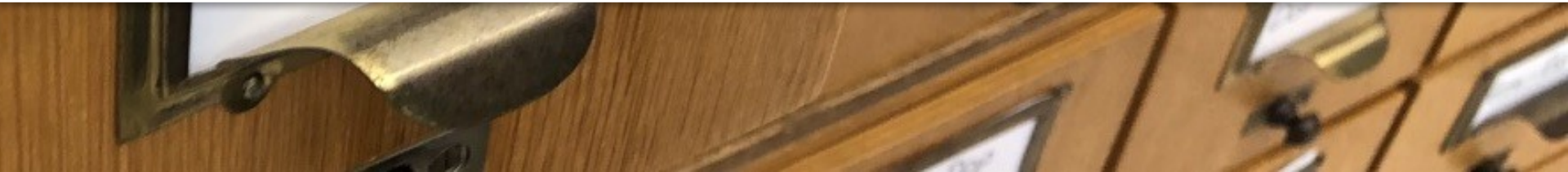


Technology
Arts Sciences
TH Köln





Teil I: Crashkurs Information Retrieval



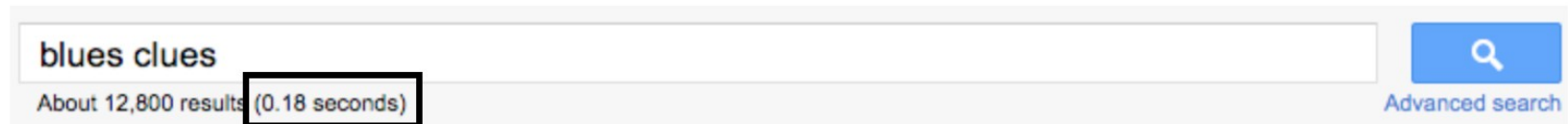
Was ist eigentlich Retrieval?

In diesem Kurs geht es um Information **Retrieval** und wir benutzen folgende, **vorläufige Arbeitsdefinition**:

*„Gegeben eine **Anfrage** und einen **Dokumentenkorpus**, finde **relevante** Dokumente.“*

- **Anfrage**: Eine Anfrage ist die Beschreibung eines **Informationsbedürfnisses**, die an das IR-System geschickt wird. Kann natürlichsprachig oder formal (Anfragesprache) sein.
- **Korpus**: Eine Sammlung von durchsuchbaren Dokumenten / Ressourcen. In unserem Falle meistens Textdokumente.
- **Relevanz**: Befriedigung des Informationsbedürfnisses eines Benutzers.

Was gehört noch zum Thema Retrieval?



Zwei zentrale Größen

- **Effizienz:** Wir wollen das Ergebnis noch vor Feierabend (oder noch besser: in 0,18 Sekunden)
- **Effektivität:** Liefere nur Ergebnisse, die Informationsbedürfnisse der Nutzer befriedigen
- Später: Klarer Fokus auf das Thema Effektivität.
- Allerdings: Wir werden uns auch darüber unterhalten, wie Suchmaschinen Ergebnisse möglichst schnell liefern können.

Effizienz von Suche: grep

- Lorem ipsum dolor sit amet,
- consetetur sadipscing elitr,
- sed diam nonumy eirmod tempor,
- invidunt ut labore et dolore magna
- aliquyam erat, sed diam voluptua.
- At vero eos et accusam et justo
- duo dolores et ea rebum.

Finde alle Zeilen mit „et“.

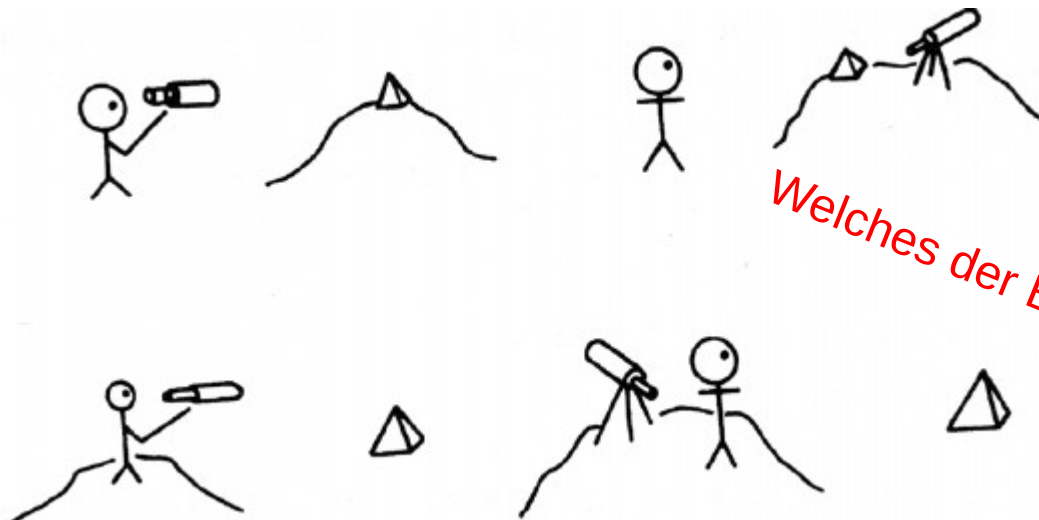
- Wie viele Schritte braucht grep?
- Ist das ein guter Weg zu suchen? Geht das nicht besser?

Effektivität: Was ist eigentlich Relevanz?

Schwierig...

„The man saw the pyramid on the hill with the telescope.“

- Viele Interpretationen dieses Satzes sind denkbar...



Welches der Bilder ist relevant?

Strukturierte Daten

Strukturierte Daten sind z.B. Tabellendaten

Angestellter	Boss	Gehalt
Berthold Heisterkamp	Bernd Stromberg	50000
Ulf Steinke	Bernd Stromberg	60000
Sinan Turçulu	Timo Becker	50000

- Numerische Anfragen und Exact Match sind möglich, bspw.: ***Gehalt < 60000 AND Boss = Timo Becker***
- Toll, aber meistens nicht das was wir im Information Retrieval wollen -> **Wir suchen in unstrukturierte Daten!**

Unstrukturierte Daten...



WIKIPEDIA
The Free Encyclopedia

- Main page
- Contents
- Featured content
- Current events
- Random article
- Donate to Wikipedia
- Interaction
 - Help
 - About Wikipedia
 - Community portal
 - Recent changes
 - Contact Wikipedia
- Toolbox
- Print/export
- Languages
 - Deutsch
 - Español
 - Bahasa Indonesia

Log in / create account

Article Discussion Read Edit View history

Gerard Salton

From Wikipedia, the free encyclopedia

Gerard Salton (8 March 1927 in [Nuremberg](#) - 28 August 1995), also known as Gerry Salton, was a Professor of [Computer Science](#) at [Cornell University](#). Salton was perhaps the leading computer scientist working in the field of information retrieval during his time. His group at Cornell developed the [SMART Information Retrieval System](#), which he initiated when he was at Harvard.

Salton was born Gerhard Anton Sahlmann on March 8, 1927 in [Nuremberg, Germany](#). He received a Bachelor's (1950) and Master's (1952) degree in mathematics from [Brooklyn College](#), and a Ph.D. from [Harvard](#) in [Applied Mathematics](#) in 1958, the last of [Howard Aiken](#)'s doctoral students, and taught there until 1965, when he joined [Cornell University](#) and co-founded its department of Computer Science.

Salton was perhaps most well known for developing the now widely used [Vector Space Model](#) for Information Retrieval^[1]. In this model, both documents and queries are represented as vectors of term counts, and the similarity between a document and a query is given by the cosine between the term vector and the document vector. In this paper, he also introduced [TF-IDF](#), or term-frequency-inverse-document frequency, a model in which the score of a term in the a document is the ratio of the number of terms in that document divided by the frequency of the number of documents in which that term occurs. (The concept of inverse document frequency, a measure of specificity, had been introduced in 1972 by [Karen Sparck-Jones](#)^[2].) Later in life, he became interested in automatic text summarization and analysis^[3], as well as automatic hypertext generation^[4]. He published over 150 research articles and 5 books during his life.


Salton was editor-in-chief of the [Communications of the ACM](#) and the [Journal of the ACM](#), and chaired [SIGIR](#). He was an associate editor of the [ACM Transactions on Information Systems](#). He was an [ACM Fellow](#) (elected 1995), received an Award of Merit from the [American Society for Information Science](#) (1989), and was the first recipient of the [SIGIR Award](#) for outstanding contributions to study of information retrieval (1983) -- now called the [Gerard Salton Award](#).

References

[\[edit\]](#)

- ^[1] G. Salton , A. Wong , C. S. Yang, A vector space model for automatic indexing [↗](#), Communications of the ACM, v.18 n.11, p.613-620, Nov. 1975
- ^[2] Spärck Jones, Karen (1972), "A statistical interpretation of term specificity and its application in retrieval" [↗](#), *Journal of Documentation* **28** (1): 11–21, doi:10.1108/eb026526 [↗](#)

Das Dokument ist teilstrukturiert...



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
 Interaction
[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact Wikipedia](#)
 Toolbox
[Print/export](#)
 Languages
[Deutsch](#)
[Español](#)
[Bahasa Indonesia](#)

[Article](#)
[Discussion](#)
[Read](#)
[Edit](#)
[View history](#)

Gerard Salton

From Wikipedia, the free encyclopedia

Gerard Salton (8 March 1927 in [Nuremberg](#) - 28 August 1995), also known as Gerry Salton, was a Professor of [Computer Science](#) at [Cornell University](#). Salton was perhaps the leading computer scientist working in the field of information retrieval during his time. His group at Cornell developed the [SMART Information Retrieval System](#), which he initiated when he was at Harvard.

Salton was born Gerhard Anton Sahlmann on March 8, 1927 in [Nuremberg, Germany](#). He received a Bachelor's (1950) and Master's (1952) degree in mathematics from [Brooklyn College](#), and a Ph.D. from [Harvard](#) in [Applied Mathematics](#) in 1958, the last of [Howard Aiken](#)'s doctoral students, and taught there until 1965, when he joined [Cornell University](#) and co-founded its department of Computer Science.

Salton was perhaps most well known for developing the now widely used [Vector Space Model](#) for Information Retrieval^[1]. In this model, both documents and queries are represented as vectors of term counts, and the similarity between a document and a query is given by the cosine between the term vector and the document vector. In this paper, he also introduced [TF-IDF](#), or term-frequency-inverse-document frequency, a model in which the score of a term in the a document is the ratio of the number of terms in that document divided by the frequency of the number of documents in which that term occurs. (The concept of inverse document frequency, a measure of specificity, had been introduced in 1972 by [Karen Sparck-Jones](#)^[2].) Later in life, he became interested in automatic text summarization and analysis^[3], as well as automatic hypertext generation^[4]. He published over 150 research articles and 5 books during his life.

Salton was editor-in-chief of the [Communications of the ACM](#) and the [Journal of the ACM](#), and chaired [SIGIR](#). He was an associate editor of the [ACM Transactions on Information Systems](#). He was an [ACM Fellow](#) (elected 1995), received an [Award of Merit](#) from the [American Society for Information Science](#) (1989), and was the first recipient of the [SIGIR Award for outstanding contributions to study of information retrieval](#) (1983) -- now called the [Gerard Salton Award](#).

References

[\[edit\]](#)

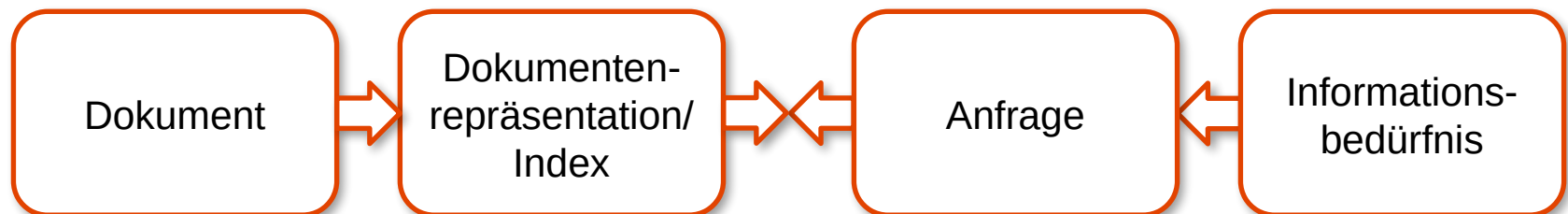
- [↑] G. Salton , A. Wong , C. S. Yang, A vector space model for automatic indexing [↗](#), Communications of the ACM, v.18 n.11, p.613-620, Nov. 1975
- [↑] Spärck Jones, Karen (1972), "A statistical interpretation of term specificity and its application in retrieval" [↗](#), *Journal of Documentation* **28** (1): 11–21, doi:10.1108/eb026526 [↗](#)

Klassisches Information Retrieval-Modell

Das klassische **Ad-Hoc-Retrieval** basiert auf Abgleich von

- Dokumenttermen (Document Representation) und
- Anfragetermen (Query).

Im klassischen Information Retrieval-Modell sind das Informationsbedürfnis als auch die Anfrage starr und verändern sich nicht.



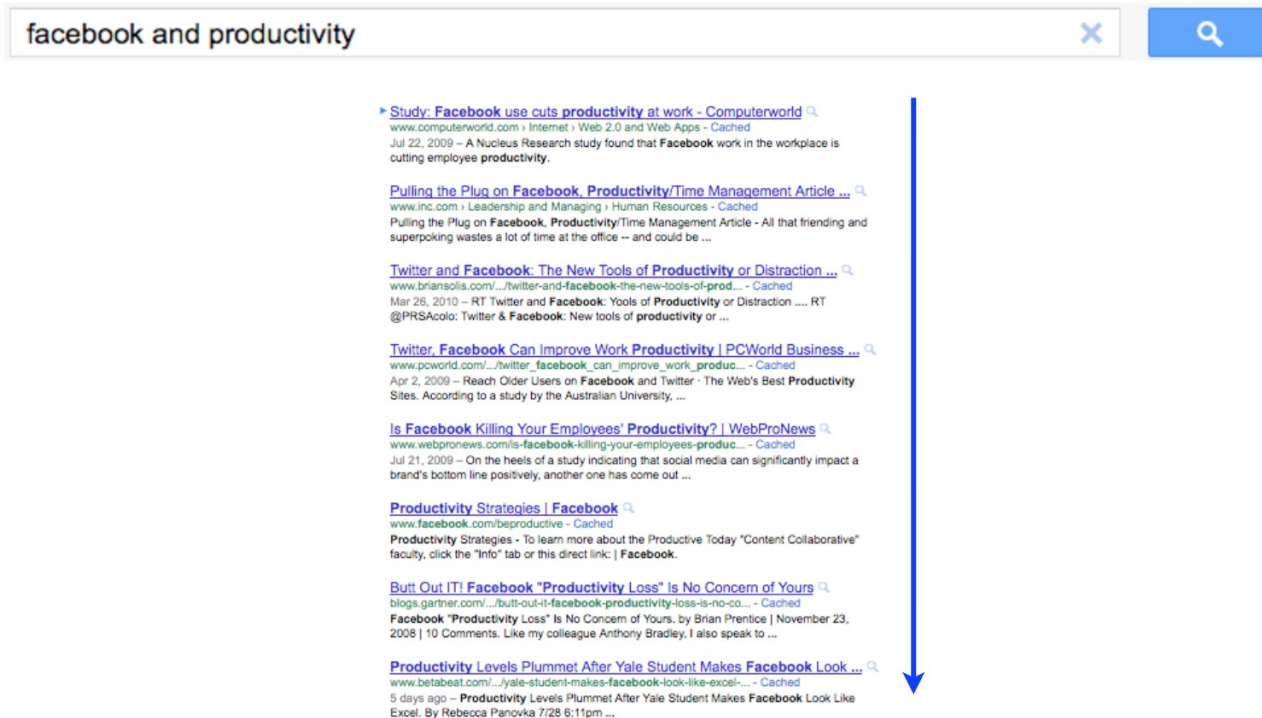
Binär

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calphurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0

Jedes Dokument ist durch einen **binären Vektor** (besteht nur aus 0/1) repräsentiert, **der** **vorberechnet wurde!**

1 wenn **Stück** das **Wort** enthält, ansonsten 0

Retrieval als Suchaufgabe



- **Ausgabe:** Ein **Ranking** von Dokumenten, in absteigender Reihenfolge Ihrer **geschätzten Relevanz** (macht es einfacher!).
- **Annahme:** Der Benutzer schaut sich die **ersten paar Dokument** an und ist zufrieden, wenn er etwas passendes gefunden hat.

Binär -> Häufigkeiten

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	157	73	0	0	0	1
Brutus	4	157	0	2	0	0
Caesar	232	227	0	2	1	0
Calphurnia	0	10	0	0	0	0
Cleopatra	57	0	0	0	0	0
mercy	2	0	3	8	5	8
worser	2	0	1	1	1	5

Jedes Dokument ist durch einen Vektor der **Termhäufigkeiten** repräsentiert.

Termfrequenz tf

- Die **Termfrequenz** (*term frequency*, *Termhäufigkeit*) $tf_{t,d}$ eines Terms t in Dokument d ist die **Häufigkeit von t in d** .
- Wir möchten die Dokumente anhand Ihres Scores, der die Übereinstimmung von Anfrage und Dokument beschreibt ranken. Hierzu wollen wir die Termfrequenz verwenden.
- Aber wie...?

Die **reinen Termfrequenzen sind ungeeignet**, weil:

- Ein Dokument mit $tf = 10$ ist sicherlich relevanter als ein Dokument mit $tf = 1$...
- Aber nicht unbedingt 10-mal relevanter...

Die **Relevanz steigt nicht proportional mit der Termfrequenz**.

Log-Termfrequenz-Gewichtung

Um die Wirkung der Termfrequenz zu dämpfen wird häufig mit der **logarithmierten Termfrequenz oder einer anderen Gewichtung** (Englisch: weight) gearbeitet:

$$w_{t,d} = \begin{cases} 1 + \log_{10}(\text{tf}_{t,d}), & \text{wenn } \text{tf}_{t,d} > 0 \\ 0, & \text{sonst} \end{cases}$$

“Das Gewicht des Terms t für das Dokument d ”

Der Score für ein Anfrage-Dokument-Paar ist die Summe über alle **Gewichtungen** aller Terme t , die sowohl in q als auch in d enthalten sind:

$$\text{Score}_{q,d} = \sum_{t \in q \cap d} w_{t,d} = \sum_{t \in q \cap d} (1 + \log_{10}(\text{tf}_{t,d}))$$

Dokumentfrequenz

Häufige Terme sind weniger informativ als seltene Terme.

- Stellen Sie sich einen Anfrageterm vor, der oft vorkommt, z.B. *hoch*, *sicher*, *teuer*...
- Ein Dokument, dass einen solchen Term beinhaltet ist wahrscheinlich relevanter, als eines das diesen Term nicht enthält. (das Grundprinzip von tf)
- Aber: Es ist kein sicherer Indikator für Relevanz.

Wir wollen **positive Gewichte** für Wörter wie *hoch*, *sicher*, *teuer*, **aber diese sollen niedriger sein als solche für seltene Terme.**

- Hierzu verwenden wir die **Dokumentfrequenz (df)**.

idf-Gewichtung

df_t ist die **Dokumentfrequenz** für t : Die Anzahl der Dokumente, die t enthält.

- df_t ist ein Maß für den **inversen Informationsgehalt** von t .
- $df_t \leq N$ (N ist die Anzahl aller Dokumente)

Wir definieren **idf** (inverse Dokumentfrequenz) von t als:

$$idf_t = \log_{10}\left(\frac{N}{df_t}\right)$$

Wir verwenden $\log_{10}(N/df_t)$ anstelle von N/df_t um den Effekt von idf zu dämpfen.

tf-idf-Gewichtung

Die **tf-idf-Gewichtung** von Termen ist das **Produkt der tf- und idf-Werte**:

$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} * \log_{10} \left(\frac{N}{\text{df}_t} \right)$$

tf-idf ist **DAS bekannteste Gewichtungsschema** im IR.

- Beachten Sie: Das „-“ ist ein Bindestrich, kein Minus.
- Andere Schreibweisen: tf.idf, tf x idf, tf*idf, TF*IDF, etc...
- Andere Kombinationen möglich... (z.B. mit diversen Logarithmen)

Der tf-idf-Wert steigt an für

- die Anzahl der **Termhäufigkeiten** in Dokumenten und
- die **Seltenheit** eines Terms in der Kollektion.

Binär -> Häufigkeiten -> Gewichte

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	4,73	2,20	0	0	0	0,03
Brutus	0,12	4,73	0	0,06	0	0
Caesar	4,09	4,00	0	0,04	0,02	0
Calphurnia	0	0,78	0	0	0	0
Cleopatra	4,44	0	0	0	0	0
mercy	0,02	0	0,02	0,06	0,04	0,06
worser	0,02	0	0,01	0,01	0,01	0,04

Jedes Dokument wird nun
repräsentiert durch eine Vektor
mit **tf-idf-Gewichten** $\in \mathbb{R}^M$

Berechnung Score pro Dokument

$$Score(q, d) = \sum_{t \in q \cap d} \text{tf-idf}_{t,d}$$

Es gibt viele, sehr viele Varianten, wie

- tf berechnet wird (mit oder ohne Logarithmus)
- die Terme in der Anfrage gewichtet werden, und, und, und...



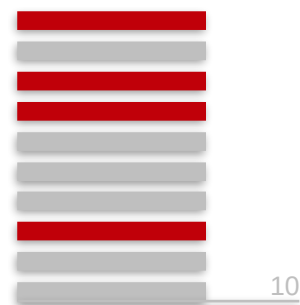
We
know you
don't like it but

IT'S
QUIZ TIME!

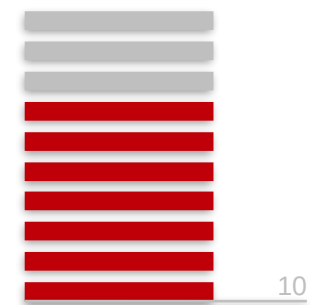
*An dieser Stelle könnten wir bereits Dokumente
ranken... Wie?*

Ranking – Welches ist besser?

System A



System B



- Relevante Treffer sind rot markiert.

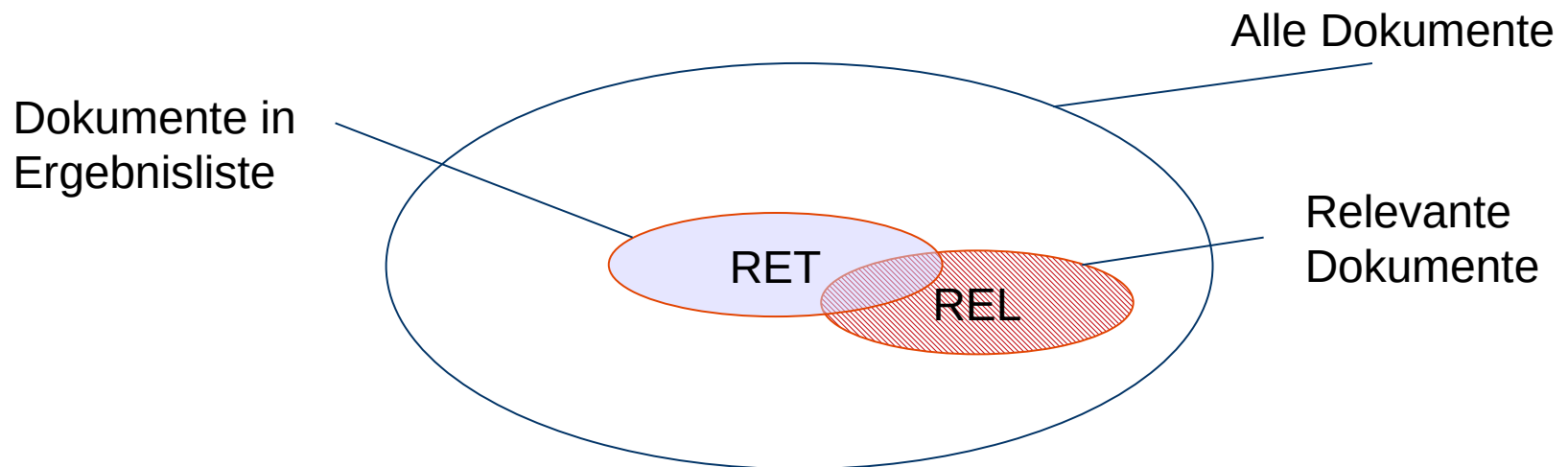
Maßzahlen für die Evaluation

- **Precision** (Treffergenauigkeit)

$$\mathcal{P} = \frac{|\text{RET} \cap \text{REL}|}{|\text{RET}|}$$

- **Recall** (Treffervollständigkeit)

$$\mathcal{R} = \frac{|\text{RET} \cap \text{REL}|}{|\text{REL}|}$$



Precision und Recall: Ein Beispiel

	Relevant	Nicht relevant
Gefunden	30	12
Nicht gefunden	14	44

- Precision $P = 30 / (30 + 12) \approx 0,714$
- Recall $R = 30 / (30 + 14) \approx 0,681$

Warum ist IR eine schwierige Aufgabe?

Information Retrieval ist ein Prozess mit **Unsicherheiten**...

- Benutzer **wissen nicht was sie eigentlich wollen**
- Benutzer wissen nicht, wie sie das was sie suchen **ausdrücken sollen**
- Computer können Nutzer keine **Kontextinformationen** entlocken, wie es z.B. ein menschlicher Bibliothekar könnte
- Computer verstehen **keine natürliche Sprache**
- Suchmaschinen müssen **erraten, was relevant ist**
- Suchmaschinen müssen erraten, wann ein Benutzer **zufrieden** ist
- ...