# Web-basiertes Question Answering

Alina Fastowski

09-06-21

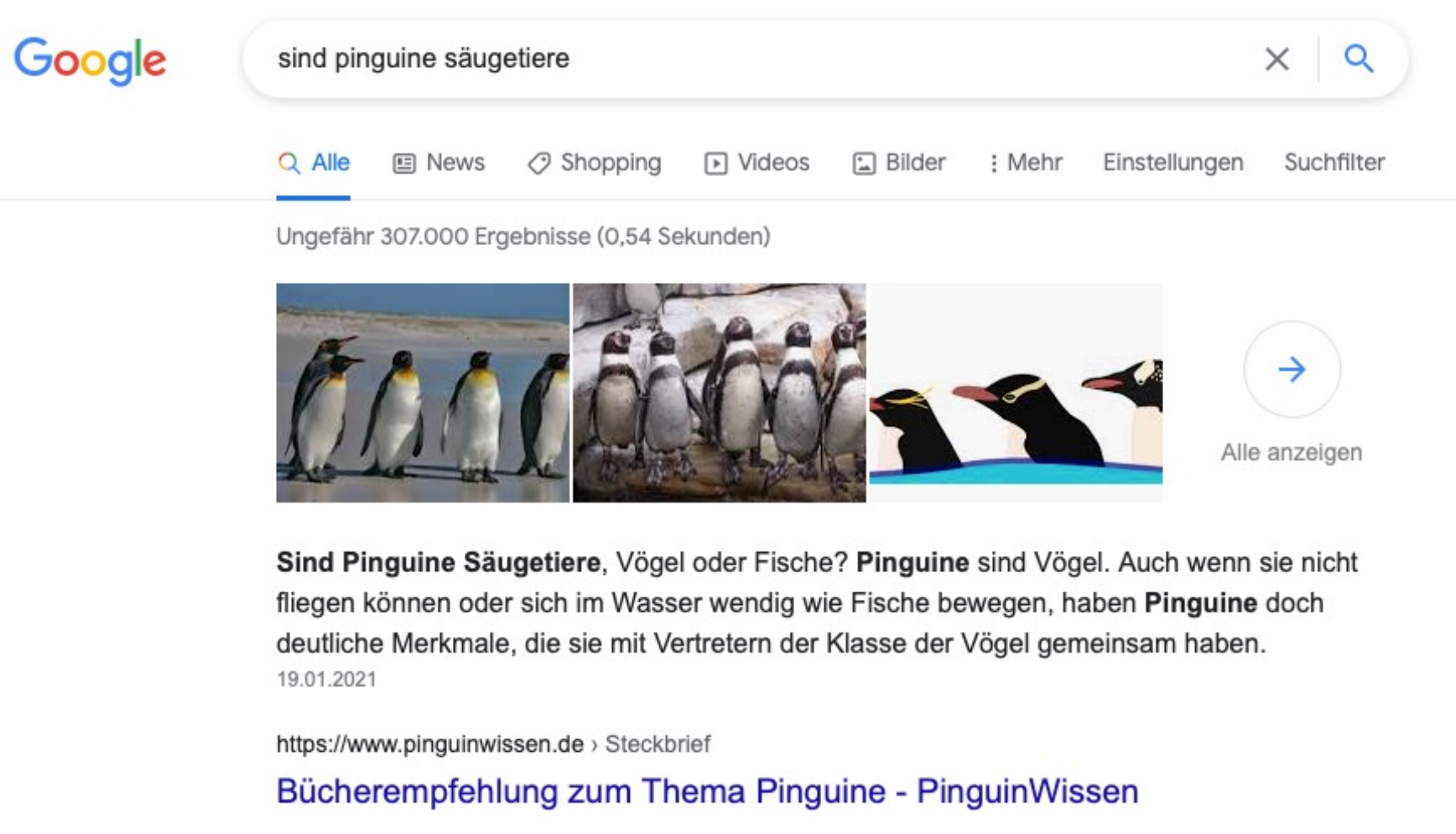
Seminar Suchmaschinen

Dozent: Stefan Langer

# Warum QA?

Sind Pinguine Säugetiere?

Pinguine sind Vögel.



# Agenda

1) Arten von QA Systemen
2) Architektur
3) Web-Indexing
4) Offene Fragen

# Arten von QA-Systemen

**Closed Domain QA**

Kann nur Fragen bestimmter Domänen beantworten, z.B. Musik, Medizin,…

Bezieht Informationen aus domänenspezifischer Ontologie.

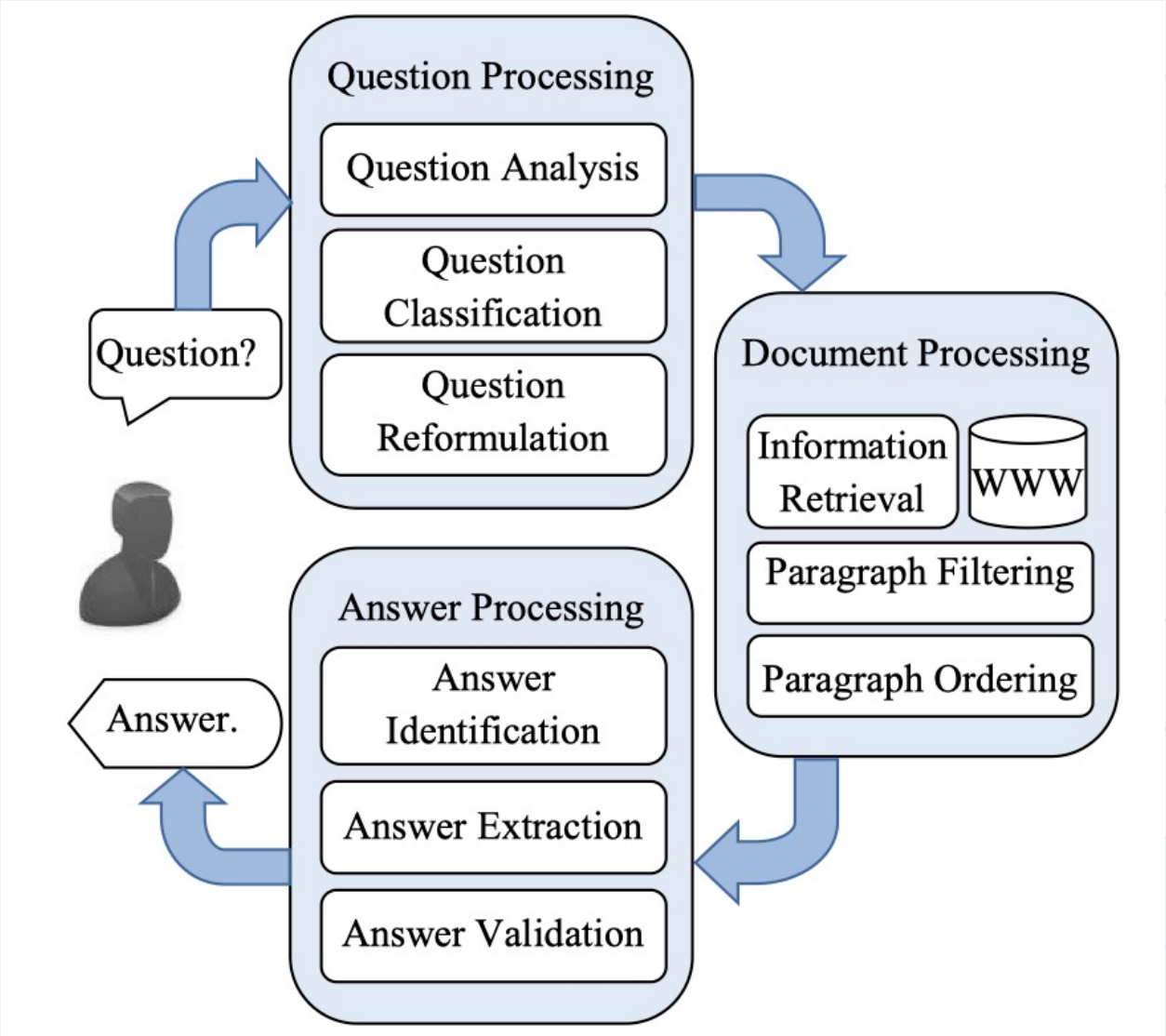
**Open Domain QA**

Kann (theoretisch) alle möglichen Fragen beantworten.

Bezieht Informationen aus universeller Ontologie oder dem WWW.

# Architektur

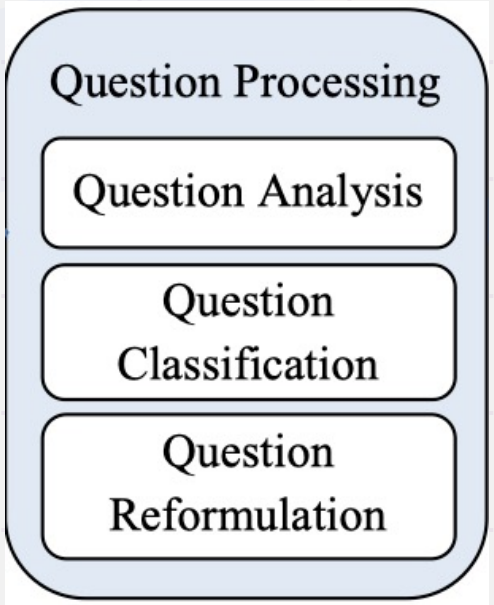
- Question Processing
- Document Processing
- Answer Processing




The Question Answering Systems: A Survey. Allam and Haggag 2012. Page 2.

# Architektur 1: Question Processing

*Ziel: Frage verarbeiten, angefragte Information verstehen.*




**Question Analysis**

Wo ist der Fokus?
*“Was ist der längste Fluss in Bayern“ -> „längste Fluss“*
Kann z.B. via pattern matching Regeln ermittelt werden.

**Question Classification**

Welche Art von Information wird angefragt?
(Dazu gleich mehr)

**Question Reformulation**

Ermittle Keywords zur Übergabe an die IR Komponente.
Verwendung von z.B. NER, Stopwords, POS-Tagger, …

Auch Verwendung von Synonymen um mehr Quellen zu finden, die die Antwort enthalten könnten.

# Architektur 1: Question Processing

**Zu Question Classification**

- Klassifiziere Typ der Frage: *was, wer, wo, wann, …*

- Klassifiziere Typ der erwarteten Antwort.

  z.B. *„ person“, „organization“* etc.

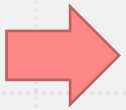
  Aber auch: „Wer spielt die Hauptrolle in XY“ -> *„actor“*

# Architektur 1: Question Processing

**(Zusätzliche mögliche Information)**

- Faktoide Fragen

-> Antwort ist kurz, ein Fakt

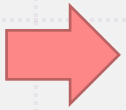


*„Was ist die Hauptstadt von Deutschland?“*

*- „Berlin“*

- Listen-Fragen

-> mehrere Antworten, Aufzählung

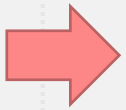


*„Welche Harry Potter Filme gibt es?“*

*-„Harry Potter und der Stein der Weisen (2001)“*
*„Harry Potter und die Kammer des Schreckens (2002)“*
*( … )*

- Definitions-Fragen

-> fordern eher ausführliche Antworten



*„Wie funktioniert ein PCR Test?“*

*- Beim PCR-Test handelt es sich um ein Standardverfahren in der Diagnostik von Viren. Der Test beruht auf der sogenannten Polymerase-Kettenreaktion (...)*

Question Processing hat ermittelt:

- Fokus der Frage
- Art der Frage
- Art der Antwort
- Keywords für die Anfrage

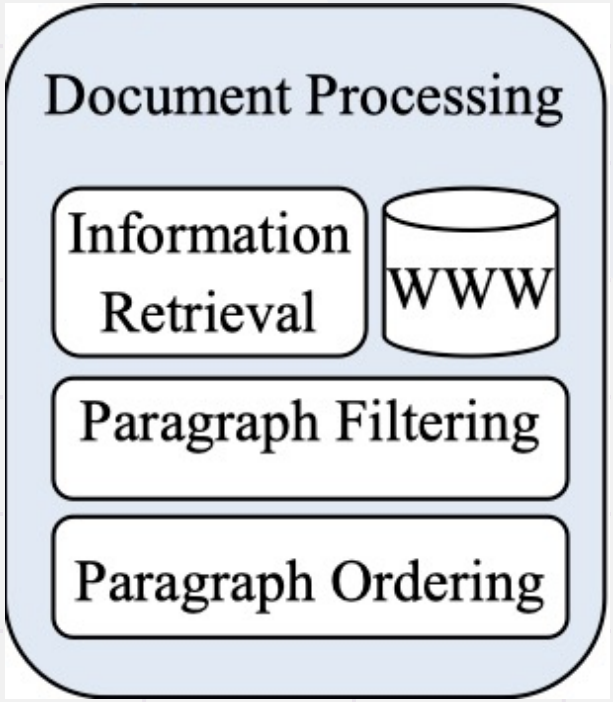
**Nächster Schritt: Document processing**

# Architektur 2: Document Processing

*Ziel: Liste an relevanten Paragraphen identifizieren*

**Information Retrieval**
Ermittle Dokumente, die relevant für die Frage sind, und ranke diese.

Query **=>** tf*idf **=>** truncate (begrenze #Dokumente)
**=>** gerankte Liste an relevanten Dokumenten



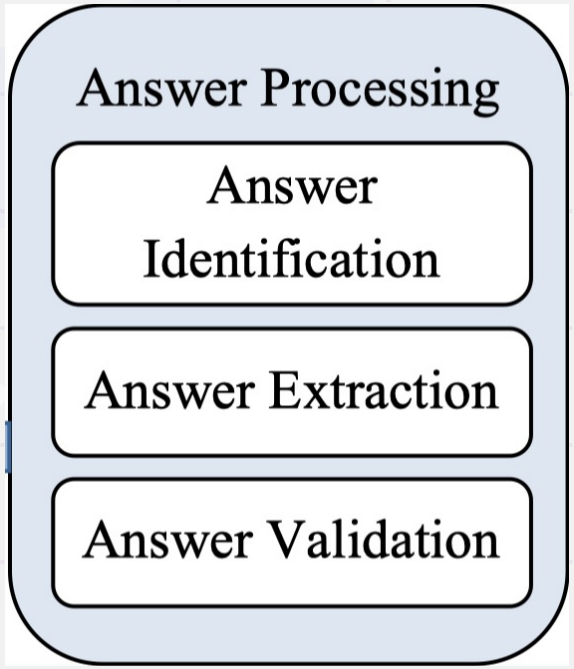

**Paragraph Filtering**
„Die relevantesten Dokumente sollten die Keywords in N benachbarten Paragraphen enthalten“ => Diese Paragraphen werden dann zurückgegeben. Der Rest des Dokuments verfällt.

**Paragraph Ordering**
Ranke die Paragraphen:

1) #Wörter aus der Frage, die genau so im Paragraphen auftreten
2) #Wörter, die die am weitesten von einander entfernten Keywords trennen
3) #Keywords, die im Paragraphen fehlen

# Architektur 3: Answer Processing

*Ziel: Generiere die endgültige Antwort*




**Answer Identification**

Ermittle Antwortkandidaten aus dem vorher ermittelten Antworttyp:
Typ "*Person*" -- NER/POS-Tagging --> Antwortkandidat(en)

**Answer Extraction**

Weitere Heuristiken (z.B. Vorkommen von Keywords), um nur die relevanten Wörter aus den Kandidaten zu ermitteln.
Hier kann stattdessen auch entschieden werden, den ganzen Paragraphen zurückzugeben.

**(Answer Validation)**

Verwendet während dem Aufbau des Systems, um den Output zu evaluieren. Dafür gibt es von Paper zu Paper verschiedene Ansätze und ist abhängig vom gesamten Systemdesign.

# Zusammenfassend:

**Question Processing**

- **Fragentyp** (was, wer, wo, …)
- **Antworttyp** (person, org, actor, book, ….)
- **Query** bestehend aus Keywords

**Document Processing**

- **Paragraphen** gefunden von Query

**Answer Processing**

- **Antwort** aus dem Paragraphen, die mit dem Antworttyp matched

# “Web-basiert“ …?

*Wie werden aus Websites Suchergebnisse? Am Beispiel von Google.*

**1) Crawling**

Welche Seiten existieren im Web?

-> Anfang: Liste an bekannten Websites aus früheren Crawlings oder Sitemaps (Einreichungen von Webseiteninhabern)

-> Crawler rufen diese auf und folgen den ausgehenden Links zu weiteren Seiten

-> Überprüfung auf: neue Websites, Änderungen an bekannten Websites, veraltete Links

Crawlings werden in bestimmten Zeitabständen durchgeführt.
*Bei einer Suche werden also nur die Seiten durchsucht,*
*die zu dieser Zeit bereits gecrawlt wurden!*

**2) Indexierung**

-> Die Inhalte einer neu gefundenen Seite werden vom Crawler geladen.

-> Jedes Wort (laut Google) erhält einen Eintrag im Index.
Wenn eine Seite indexiert wird, wird sie allen Einträgen für Wörter hinzugefügt, die auf dieser Seite vorkommen.

-> Der Google-Suchindex umfasst Milliarden von Websites und ist über 100.000.000 Gigabyte groß. (100.000 Terabyte/100 Petabyte)

*Auf diesen Index könnten wir nun ein QA-System loslassen.*

*Anmerkung: Google gibt keine detaillierten Informationen preis. Der ganze Prozess ist natürlich viel komplexer, damit die Suchmaschine so funktionieren kann, wie sie es tut.*

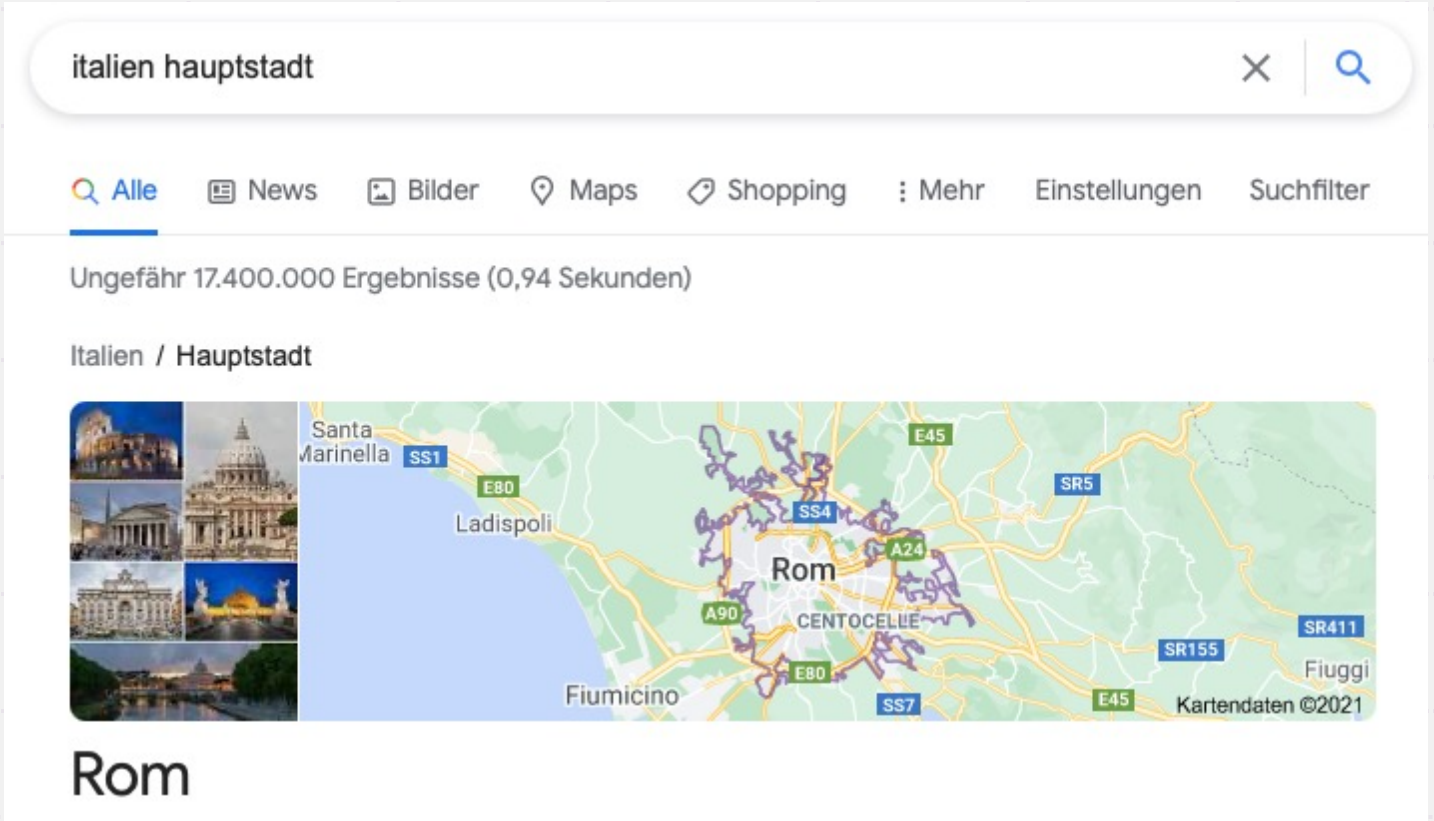
# Offene Fragen

- Wie kann QA so schnell und effizient funktionieren?
- Wie klassifizieren Suchmaschinen Fragen?

  Eine Anfrage enthält nicht unbedingt

  - Fragezeichen
  - für Frage typischen Satzbau
  - typische Fragewörter

… und trotzdem erhält man eine direkte Antwort.



# Danke für eure Aufmerksamkeit

# Quellen


- „The Question Answering Systems: A Survey“. Allam and Haggag. International Journal of Research and Reviews in Information Sciences (IJRRIS) Vol. 2, No. 3, September 2012.
- „Open-Domain Question-Answering“. John Prager. Foundations and Trends in Information Retrieval Vol. 1, No. 2 (2006) 91–231.
- „An Introduction To Question Answering Systems“. https://www.section.io/engineering-education/question-answering/
- „How Search Works“. Google. https://developers.google.com/search/docs/beginner/how-search-works