

# „Relation extraction and scoring in DeepQA“

**Statistische und regelbasierte Verfahren zur Extraktion  
und Bewertung von semantischen Relationen**

# Übersicht

**1) Einführung**

**2) Regelbasierter Ansatz**

**3) Statischer Ansatz**

**4) Integration in DeepQA**

**5) Evaluierung der Ansätze**



# 1) Einführung



# Was ist eine semantische Relation?

- Beziehung zwischen den Bedeutungen von sprachlichen Ausdrücken (Wörter, Sätze)
- Beispiel:  $\text{composerOf} \rightarrow \text{np} : [\text{Composer}] \rightarrow \text{nadj} \rightarrow [\text{Music}]$

# Extraktion von semantischen Relationen?

| <i>Jeopardy! question</i>   | <i>Relation detected (relations are from the DBpedia knowledge base)</i> |
|---|--|
| MOTHERS & SONS: Though only separated by one year in real life, she played mother to son Colin Farrell in "Alexander."      | Starring (she, "Alexander")  |
| THE DEVIL: "The Screwtape Letters" from a senior devil to an under devil are by this man better known for children's books. | Author (man, "The Screwtape Letters")                                    |
| THE LONE REPRESENTATIVE: Michael Castle from this state with 3 counties: New Castle, Kent and Sussex.                       | Residence ("Michael Castle", state)                                      |

# Bewertung von semantischen Relationen?

- Question:
  - „*This hocket defenseman ended his career on June 5, 2008*“
- Supporting passage:
  - „*On June 5, 2008, Wesley annouced his retirement after his 20th NHL season.*“



# Wo liegt das Problem? (1 / 2)

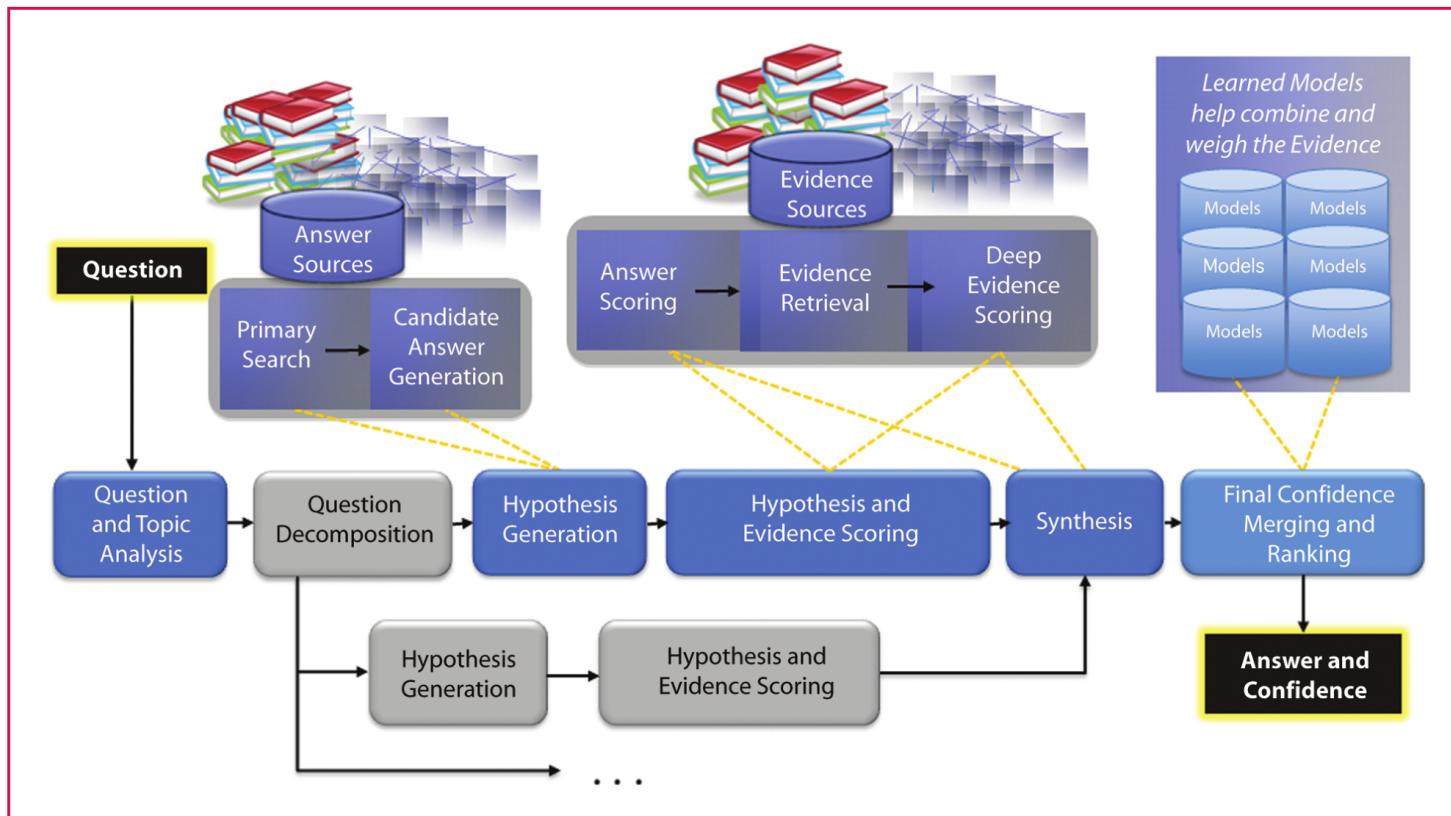
- Theoretisch unendlich viele Möglichkeiten, eine Relation lexikalisch und syntaktisch darzustellen
- Beispiel: authorOf:
  - Lexikalische Möglichkeiten: *write, compose, pen, publish...*
  - Syntaktische Möglichkeiten: *write, written, uvm.*



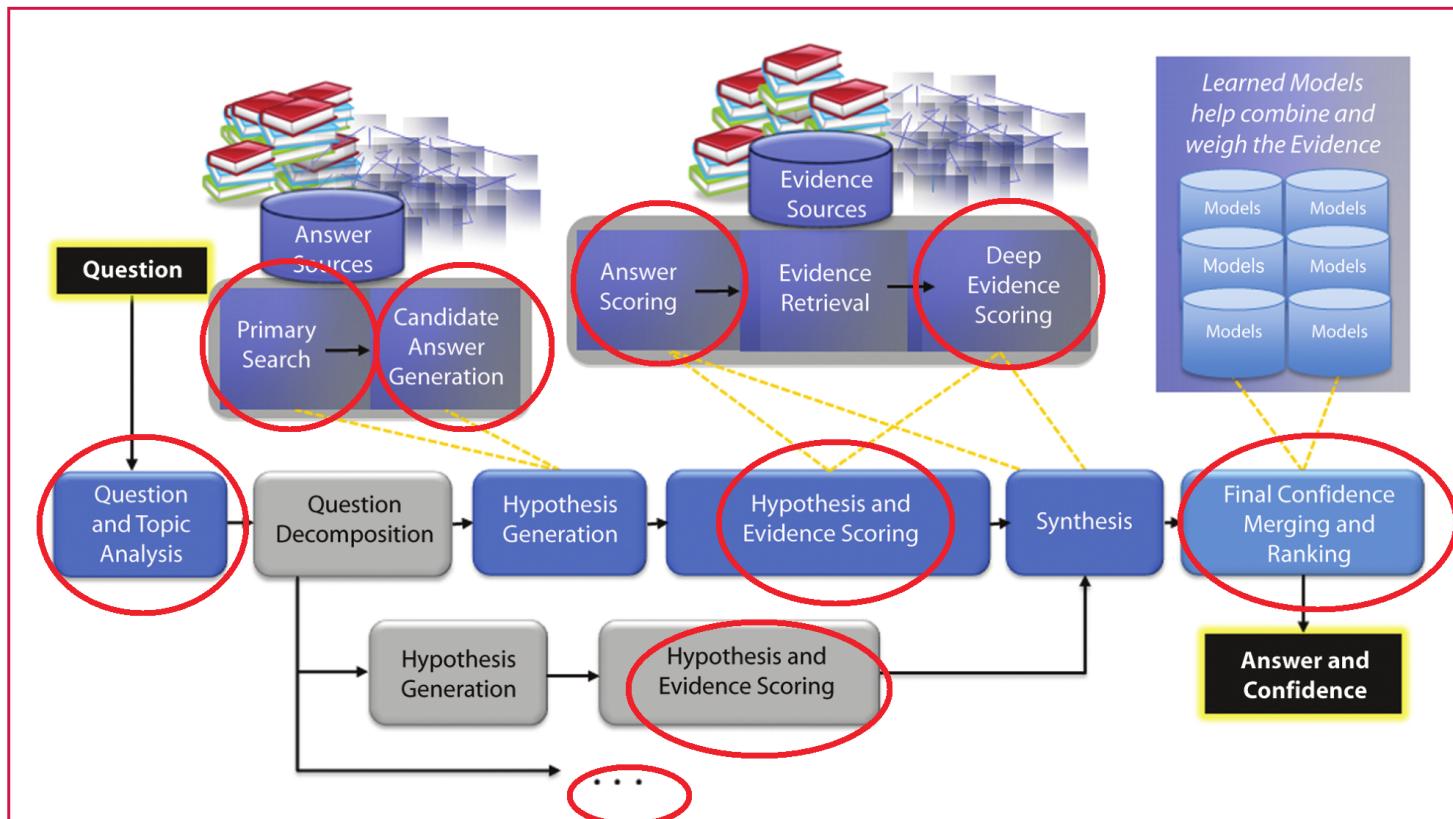
# Wo liegt das Problem? (2 / 2)

- Aber: In der Regel gibt es einen „üblichen Sprachgebrauch“, der die Anzahl an möglichen Ausdrucksweisen für eine semantische Relation begrenzt
- Daher lassen sich Regeln und Muster erkennen und formalisieren

# Wo befinden wir uns in DeepQA?



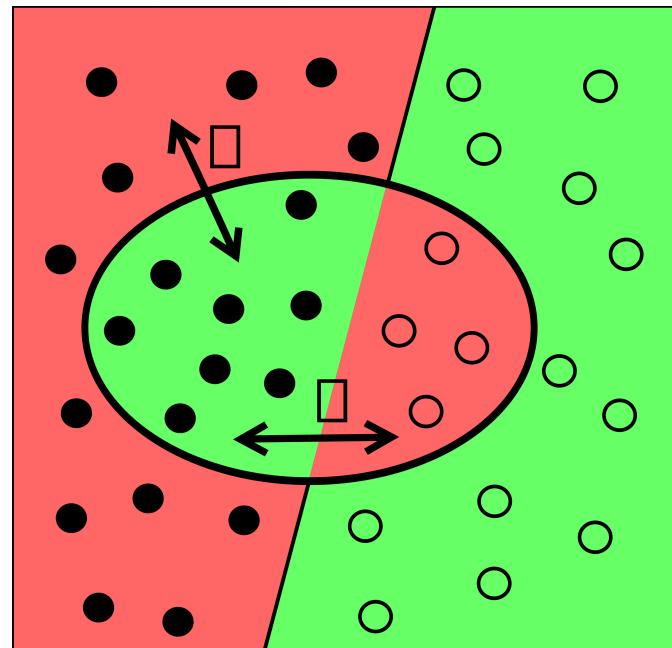
# Wo befinden wir uns in DeepQA?





# Precision und Recall

- Zwei Metriken für die Bewertung von Mustererkennungsverfahren





# Warum zwei Verfahren?

- Man möchte sowohl Precision als auch Recall maximieren
- Regelbasierte Verfahren haben in der Regel hohe Precision, sind aber aufwendig zu erstellen
- Statistische Verfahren ermöglichen einen hohen Recall, sind aber in der Regel weniger präzise

# Grundlagen

- English slot grammar (ESG) parser
- Entity disambiguation and matching (EDM)
- Predicate disambiguation and matching (PDM)
- Answer Lookup



## 2) Regelbasierter Ansatz

# Auswahl der Relationen (1/2)

- Lexikalische Analyse von historischen Jeopardy Fragen →
- Liste von Wörtern, die für eine bestimmte Domäne ein starkes Indiz für eine spezielle Semantische Relation darstellen

## Auswahl der Relationen (2/2)

- Für welche Relationen erstellt man nun manuell Regeln?
- Relationen, die besonders relevant für ihre Domäne sind
- Relationen, die domänenübergreifend besonders häufig relevant sind



# Beispiele

- *Robert Redford and this „Picket Fences“ star both debuted as soldiers in the 1962 drama „War Hunt“.*  
(actorIn, actorOf, timeStamp)
- *Born in Winsted, he practiced law in Connecticut before he wrote „Unsafe at Any Speed“.*  
(bornWhere, authorOf)
- *This Norwegian star of such movies as „Autumn Sonata“ was actually born in Japan.*  
(nationalityOf, actorIn, bornWhere)
- *The main library at the University of Northern Colorado is named for this alumnus who wrote an epic of Colorado in 1974.*  
(namedAfter, authorOf)



# Verfahren (1/2)

- Erkennen von syntaktischen Konstituenten (ESG)
- Erkennen von Abhängigkeiten und Zuordnung von Abhängigkeitsbäumen zu Predicate-argument structure (PAS)
- Ein semantisches Typensystem zur Überprüfung von semantischen Constraints

# Verfahren (2/2)

- Regeln zur Erkennung von Relationen geschrieben in Prolog
- Unification pattern matching auf der Predicate-argument structure

# Sparsamkeit bei der Regelspezifikation

- Stark typisierte Typen für Relationen
- Ausnutzung des lexikalischen Raumes der Expression von Relationen und Ähnlichkeiten in der Syntax bei dem Ausdruck von unterschiedlichen Relationen
- Metamuster: authorOf, directorOf, composerOf → creatorOf



# Anzahl an Regeln

- Das regelbasierte Verfahren erkennt ungefähr 30 verschiedene Relationen
- Zu jeder Relation gibt es durchschnittlich 10-20 Regeln



### 3) Statistischer Ansatz



# Mustererkennung

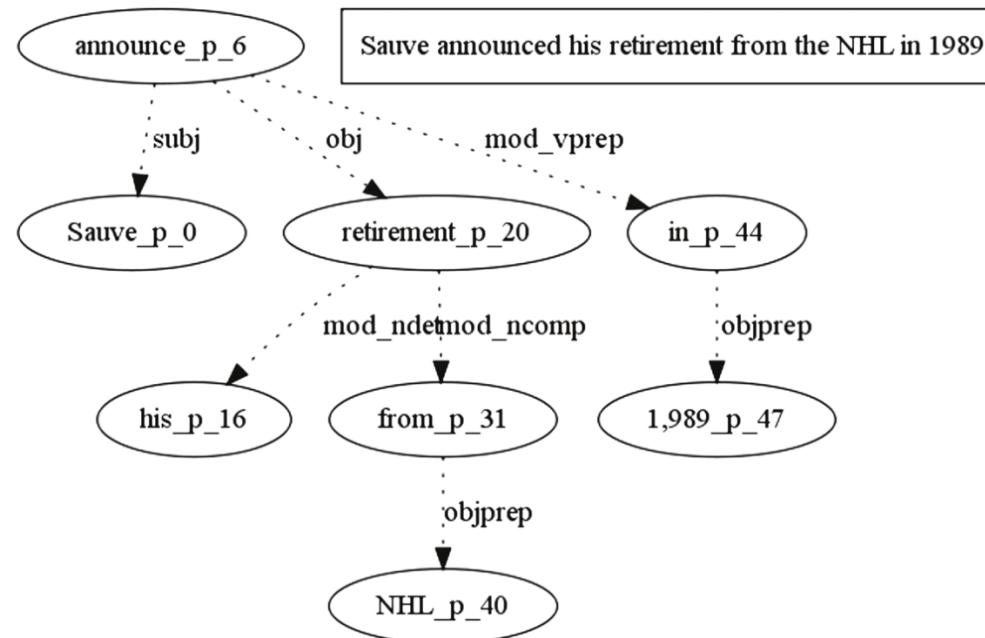




# YAGO

- „Yet Another Great Ontology“
- Automatisch aus Wikipedia und WordNet extrahierte Ontologie von semantischen Typen und semantischen Relationen
- 1.7 Millionen Einträge, 15 Millionen „Fakten“

# Abhängigkeitsbäume

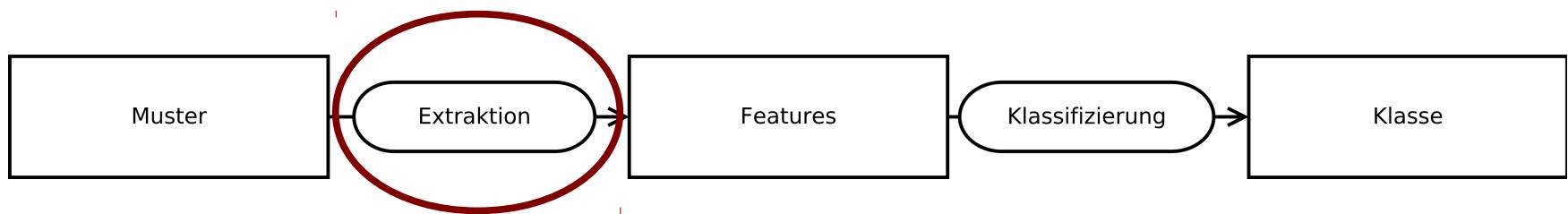




## 3.1) Repräsentation von Instanzen einer Relation



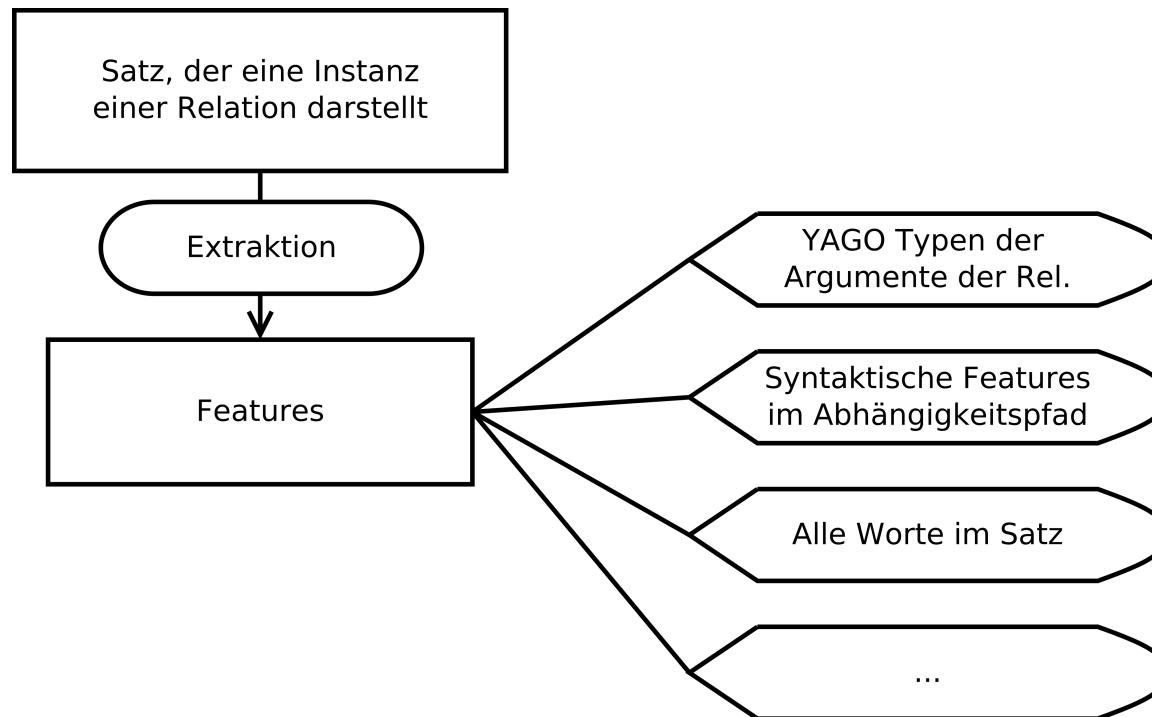
# Mustererkennung





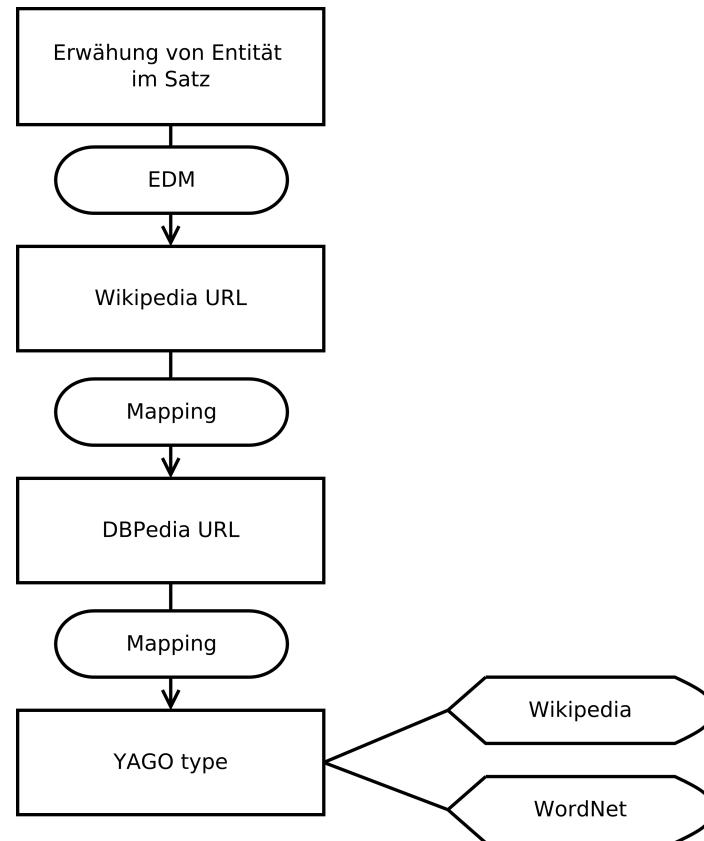
# Features - Übersicht

„Sauve announced his retirement from the NHL in 1989“

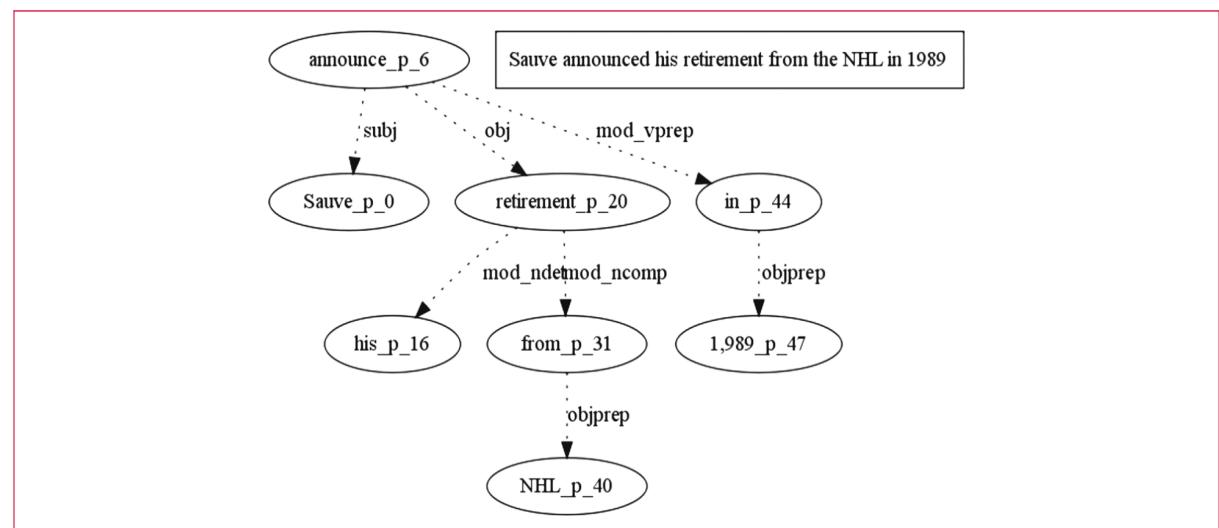
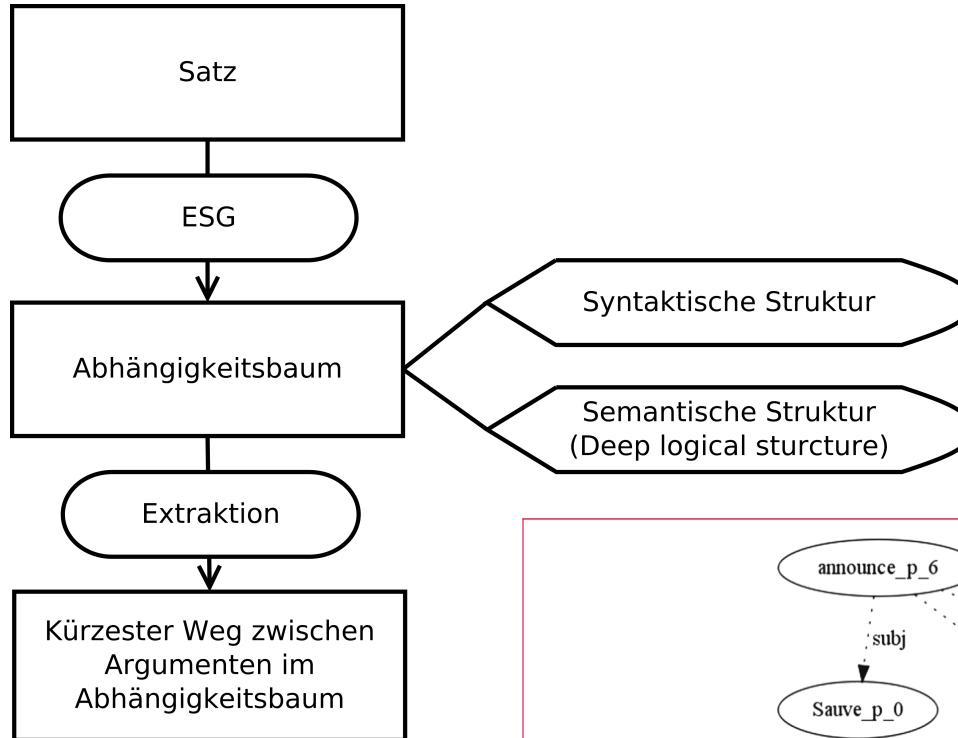




# Features – YAGO type



# Features – Abhängigkeitspfad





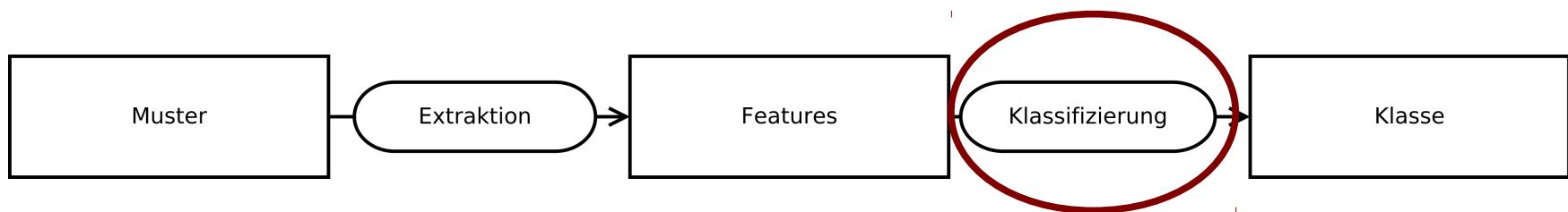
# Features - Beispiel

„Sauve announced his retirement from the NHL in 1989“

- Relation: ActiveYearEndDate
- argument1type [Person, Athlete], argument2type [Year]
- Syntaktische features im Abhängigkeitspfad: [subj, mod\_vprep, objprep]
- Alle Worte im Satz: ...



# Mustererkennung





# Was möchte man erhalten?

- Semantische Relationen
- Möglichst viele Sätze, die diese Relationen ausdrücken
- Beispiel: authorOf →
  - „Book X was written by Y“
  - „Published in 1987 by Y, X was influential...“
  - ...



## 3.2) Extraktion von Relationen aus Wikipedia



# Warum Wikipedia als Quelle? (1/2)

- DBpedia: Quelle für strukturierte Informationen mit direkter Zuordnung zu Wikipedia Infoboxen und zugehörigen Artikeln
- Dadurch einfache Zuordnung von in DBpedia definierten Relationen zu Sätzen in den entsprechenden Wikipedia Artikeln



# Warum Wikipedia als Quelle? (2/2)

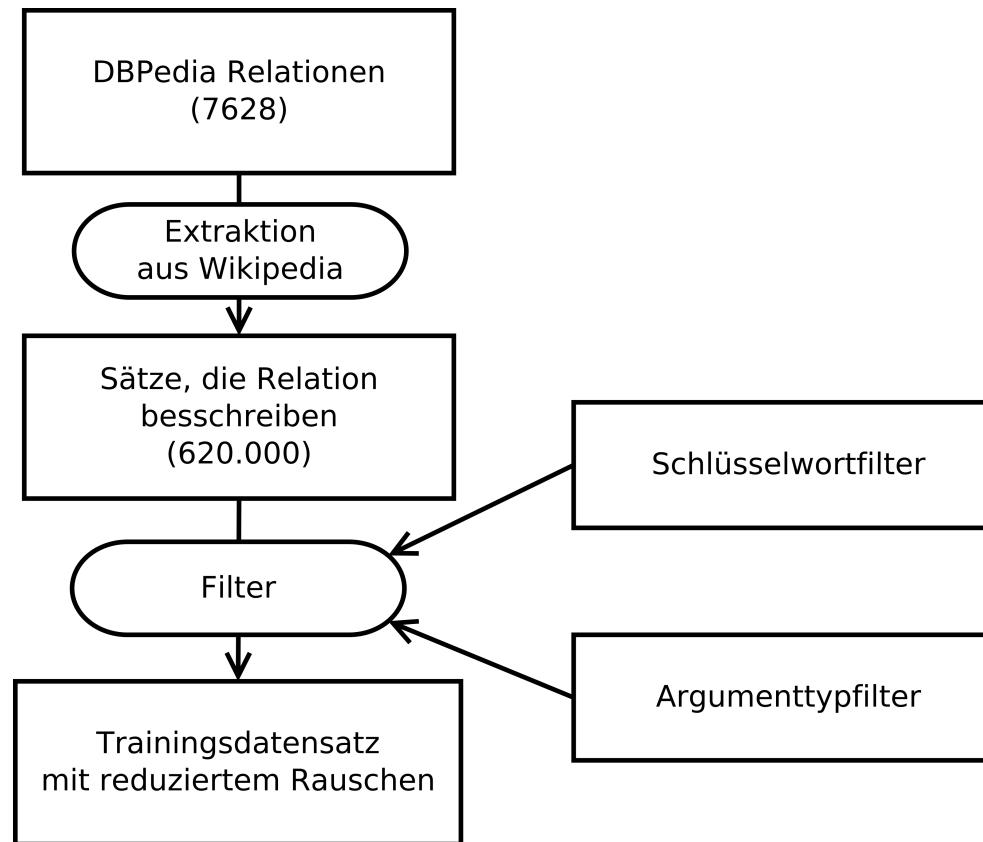
- Zwei Annahmen:
  - Eine in der Infobox beschriebene Relation findet sich auch in einem Satz im Text
  - Erster Satz im Text, der beide Argumente einer Relation enthält, beschreibt diese Relation



# Beispiel

- Wikipedia Artikel zu Albert Einstein
- Semantische Relation in Infobox: „*alma mater*“ → University of Zurich
- Erster Satz, der „*Einstein*“ und „*University of Zurich*“ enthält:
  - „*Einstein was awarded a PhD by the University of Zurich*“

# Verfahren





# Schlüsselwortfilter

- Ein Testdatum (Satz) wird nur übernommen, wenn mindestens ein Wort im Abhängigkeitspfad ist, welches:
  - In mindestens 10% der positiven Instanzen vorkommt
  - Mindestens 3 mal in den positiven Trainingsdaten vorkommt



# Argumenttypfilter

- Basiert auf der Typenhierarchie von YAGO
- Jedes Argument einer Instanz einer Relation muss einem YAGO Typ entsprechen, welches von einem YAGO Typ erbt, welcher für diese Art von Relation üblich ist



## 3.3) Relation topics



# Motivation

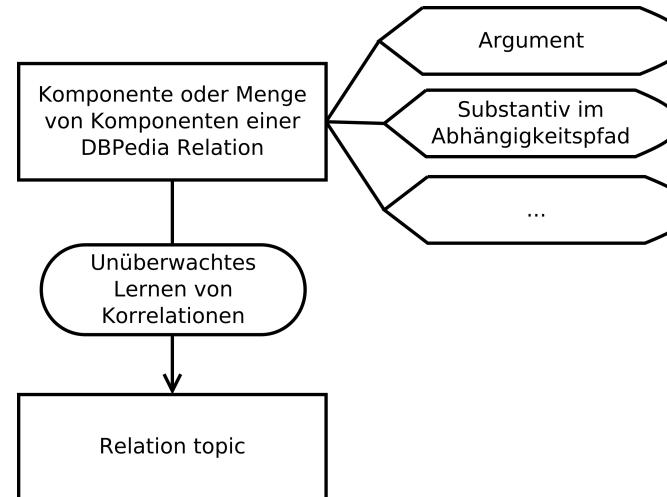
- Man sucht eine Metrik, die eine Instanz einer Relation mit den bekannten Relationen in Relation setzt
- ... unabhängig davon, ob es eine Instanz von einer bekannten oder unbekannten Relation ist



# Idee

- Man definiert relation topics als multinominale Distributionen auf den bekannten DBPedia Relationen
- Beispiel: [ doctoraladvisor (0.683366), doctoralstudents (0.113201), candidate (0.014662), academicadvisors (0.008623), notablestudents (0.003829), college (0.003021), operatingystem (0.002964), ... ]

# Umsetzung

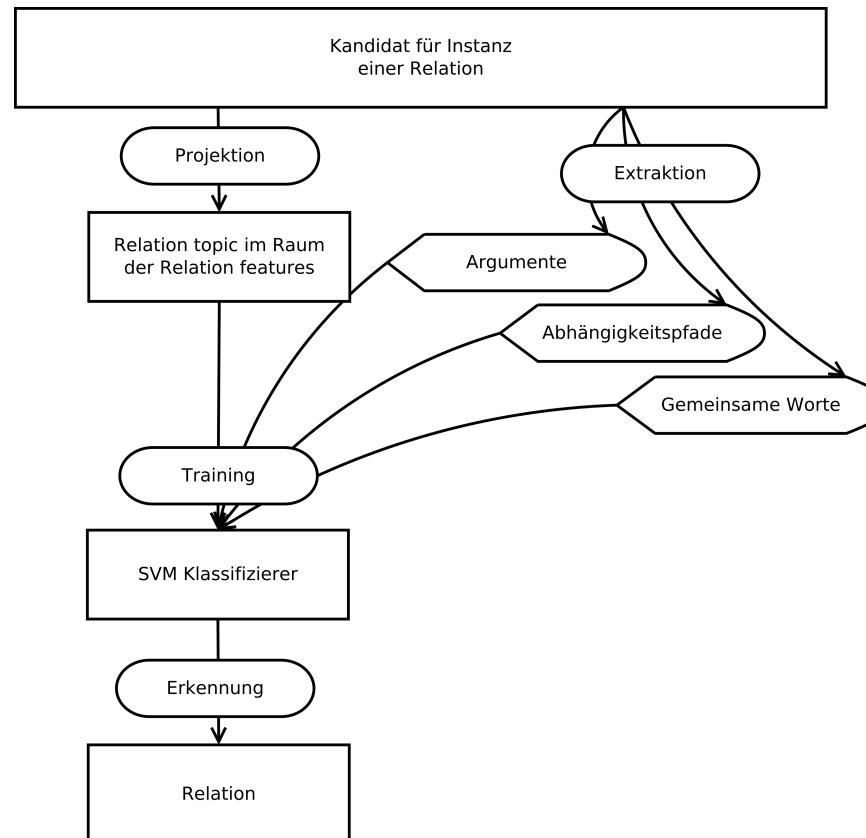


- Ein relation topic vector hat eine Länge von 7,628 (entspricht Anzahl der in DBpedia definierten Relationen)
- Relation topics sind orthonormal zueinander



## 3.3.1) Relationserkennung mit relation topics

# Verfahren



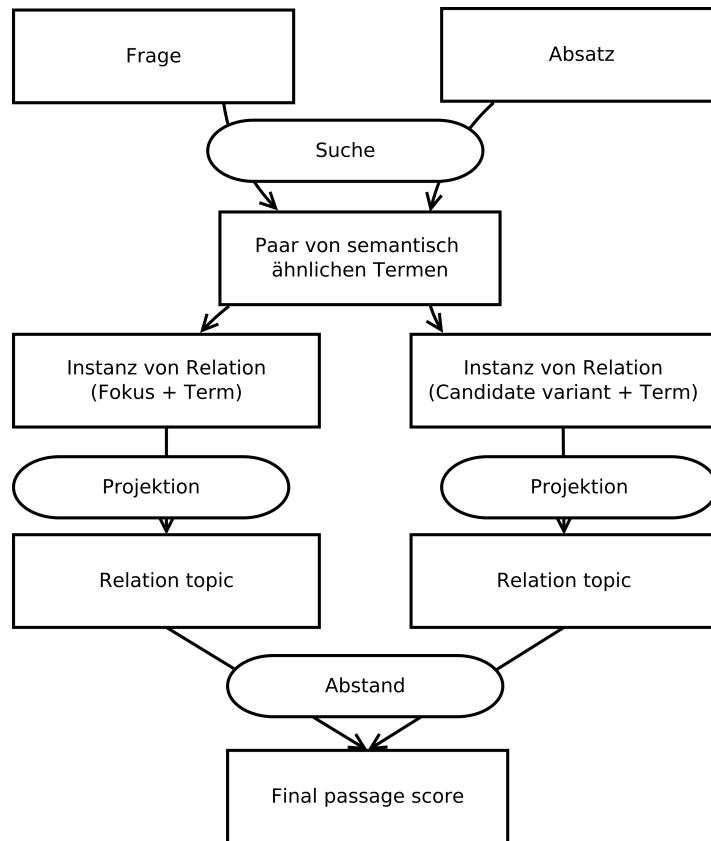
# Training

- Trainingsdaten aus Wikipedia
- Die Argumente der Relationen sind bekannt
- Die negativen Trainingsdaten werden aus den Daten für die jeweils anderen Relationen gesampelt



## 3.3.2) Relationsbewertung (Unstructured passage scoring) mit relation topics

# Verfahren



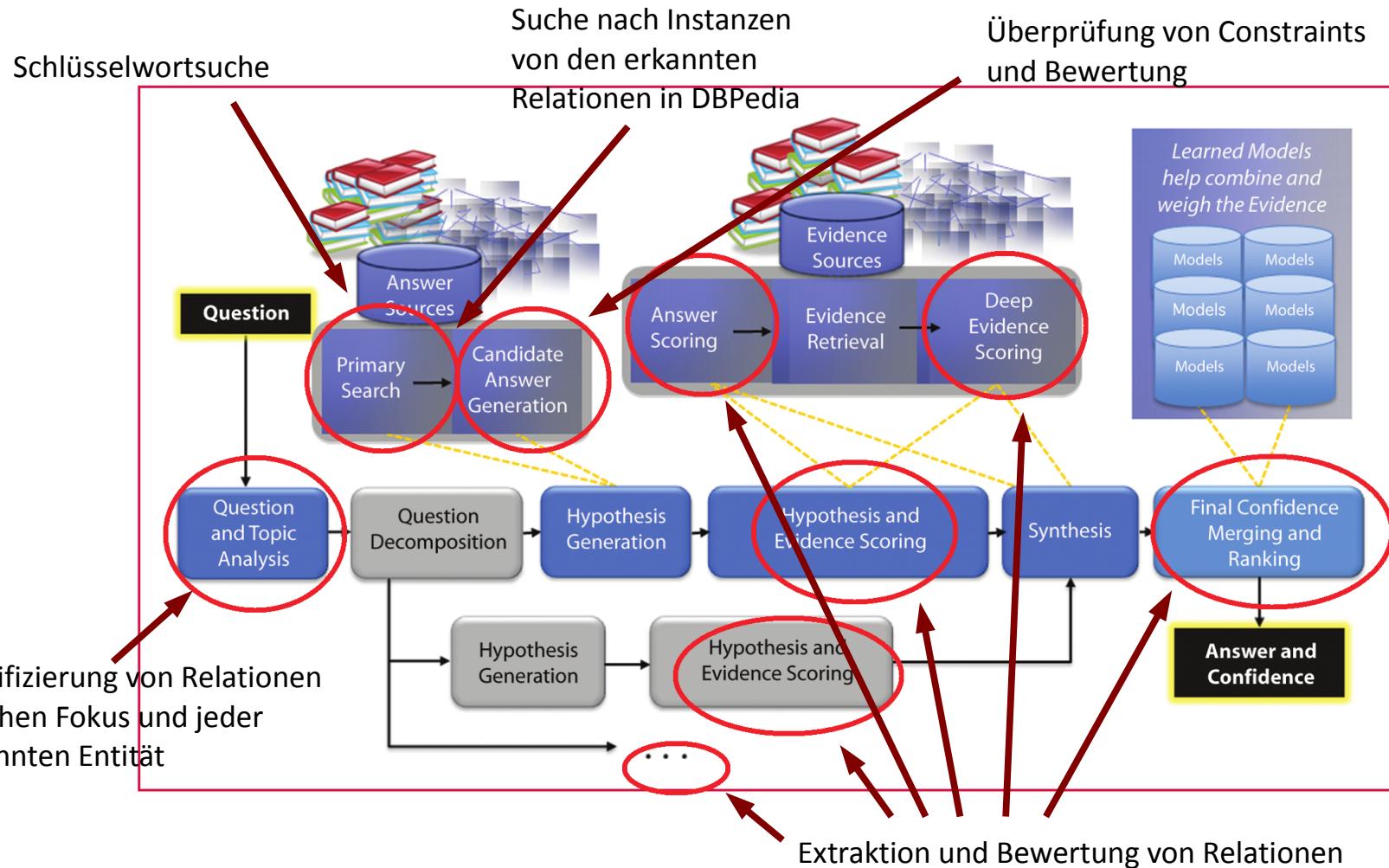
Question: *It's Mexico's northernmost state, but part of its name means „low“:*

Passage: *Cabo Catoche or Cape Catoche, in the Mexican state of Quintana Roo, is the northernmost point on the Yucatan Peninsula. It is about 53 km (33 miles) north of the city of Cancun.*



## 4) Integration in DeepQA

# Integration in DeepQA?



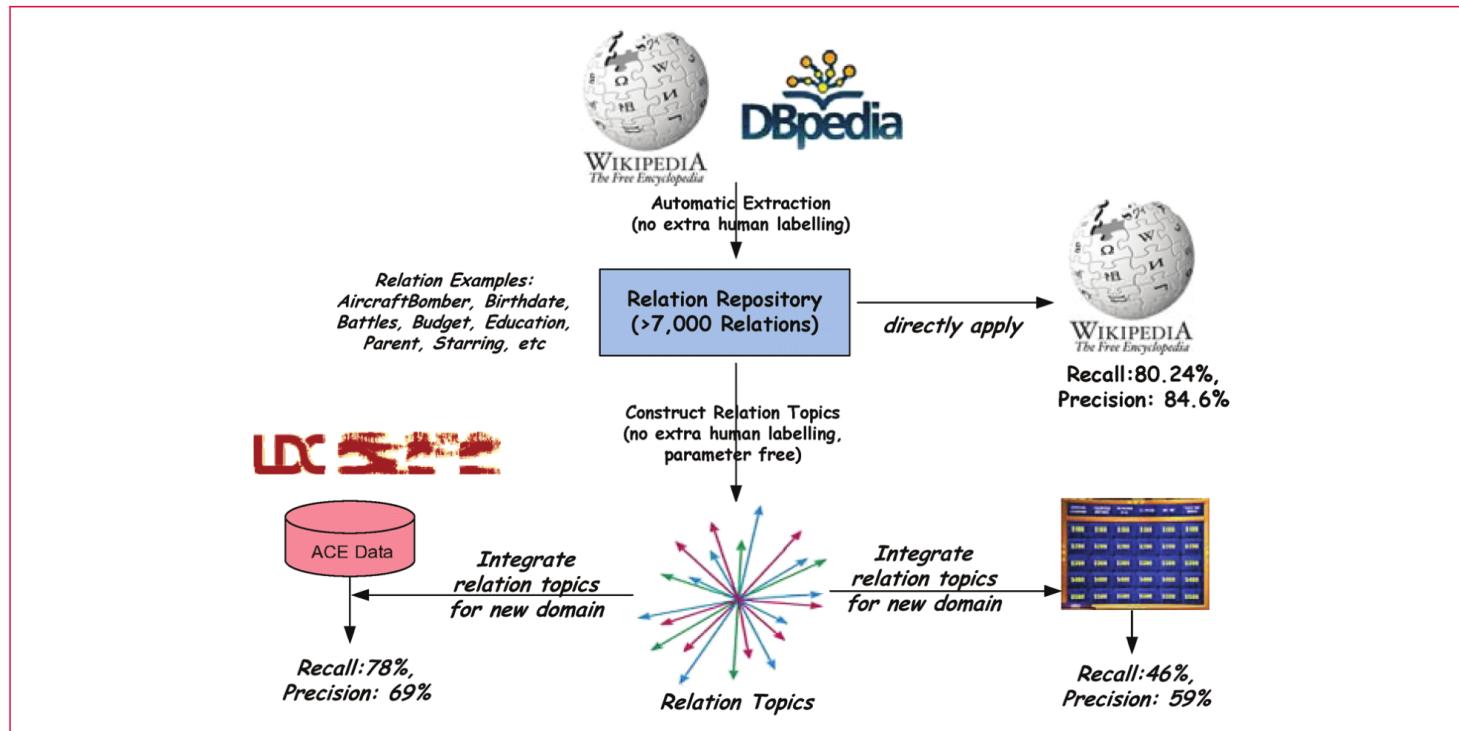


## 5) Evaluierung der Ansätze



## 5.1) Evaluierung auf Komponentenebene

# Statistische Relationserkennung



# Detaillierte Evaluierung

**Table 2** Component-level evaluation of different approaches using ACE 2004 data.

| <i>Approach</i>                | <i>Recall</i> | <i>Precision</i> | <i>F<sub>1</sub> score</i> |
|--------------------------------|---------------|------------------|----------------------------|
| Convolution tree kernel        | 56.7%         | 72.5%            | 0.636                      |
| Composite kernel (linear)      | 67.00%        | 73.5%            | 0.701                      |
| Syntactic kernel               | 70.5%         | 69.23%           | 0.6986                     |
| Nguyen et al. (2009) [30]      | 67.00%        | 76.60%           | 0.7150                     |
| Our kernel with topic features | 77.88%        | 69.15%           | 0.7324                     |

**Table 3** Component-level evaluation using Jeopardy! data.

|  | <i>No. relation detectors</i> | <i>Recall</i> | <i>Precision</i> | <i>F<sub>1</sub> score</i> |
|--|-------------------------------|---------------|------------------|----------------------------|
| Rule-based approach <sup>a</sup>       | ~30 relations                 | 41.40%        | 83.25%           | 0.5530 <sup>a</sup>        |
| Statistical approach (without filters) | >7,000 relations              | 54.28%        | 25% (estimated)  | 0.3423                     |
| Statistical approach (with filters)    | >7,000 relations              | 45.71%        | 59.44%           | 0.5168                     |
| Filters only                           | >7,000 relations              | 83.00%        | —                | —                          |

<sup>a</sup>The rule-based approach is evaluated on nine relations.



## 5.2) Evaluierung auf Systemebene

# Detaillierte Evaluierung

**Table 4** System-level impact in end–end experiments of relation detection-based candidate answer generation.

| <i>Configuration</i>                                | <i>Accuracy improvement</i> |
|---|-----------------------------|
| Basic configuration                                 | 54.36%                      |
| Basic + rule-based                                  | 54.79% (+0.43%)             |
| Basic + rule-based + statistical relation detection | 54.59% (+0.23%)             |
| Full configuration                                  | 69.81%                      |
| Full + rule-based + statistical relation Detection  | 70.35% (+0.54%)             |

**Table 5** System-level impact in end–end experiments of relation detection-based unstructured passage scoring.

| <i>Configuration</i>                             | <i>Accuracy improvement</i> |
|--|-----------------------------|
| Basic configuration                              | 54.36%                      |
| Basic + rule-based                               | 54.79% (+0.43%)             |
| Basic + rule-based + statistical passage scoring | 56.79% (+2.43%)             |
| Full configuration                               | 69.81%                      |
| Full + rule-based + statistical passage scoring  | 70.41% (+0.6%)              |