Intro to Question Answering

Manish Shrivastava

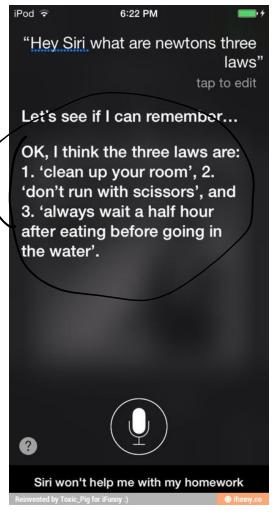
A Question on questions

- Can we ask
 - Who is Narendra Modi?
 - Where was the <u>current prime</u> minister born?
 - Where is Gujarat/ Gandhinagar ?
 - Why is rupee falling in comparison to dollar?

Question answering



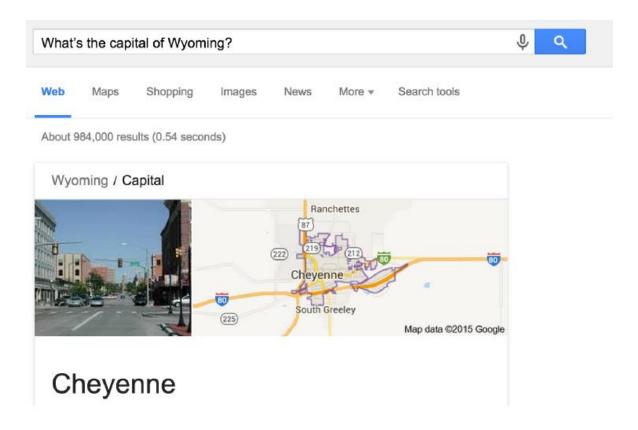
'Watson' computer wins at 'Jeopardy'



credit: ifunny.com

Question answering

Go beyond search



Go beyond traditional web search

Answer queries using structured knowledge

- Google Knowledge Graph
- Facebook Graph Search
- Bing's Satori

Need to understand the query!

The following slides are modified from Christopher Manning & Pandu Nayak's intro. to IR.

CS6501-NLP

Why is QA hard?

- Understanding that a query means a question.
- Translating the question into a sequence of tokens useful for retrieval.
- Retrieve appropriate <u>units of text</u> (vanilla IR may not work!)
- Filter, rank, and/or combine the units into an answer.
- Summarize the answer.
- Visualize it.
- Different types of question need different treatments.

Factoid Questions

- Who founded Virgin Airlines?
- What is the average age of the onset of autism?
- Where is Apple Computer based?

• Courtesy: Speech and Language Processing by Dan Jurafsky and James Martin, Chapter 23.

•

Non-factoid Questions

- Who is Celia Cruz?
- What is a Hajj?
- In children with an acute febrile illness, what is the efficacy of single-medication therapy with acetaminophen or ibuprofen in reducing fever?

• Courtesy: Speech and Language Processing by Dan Jurafsky and James Martin, Chapter 23.

Factoid vs Non-factoid QA

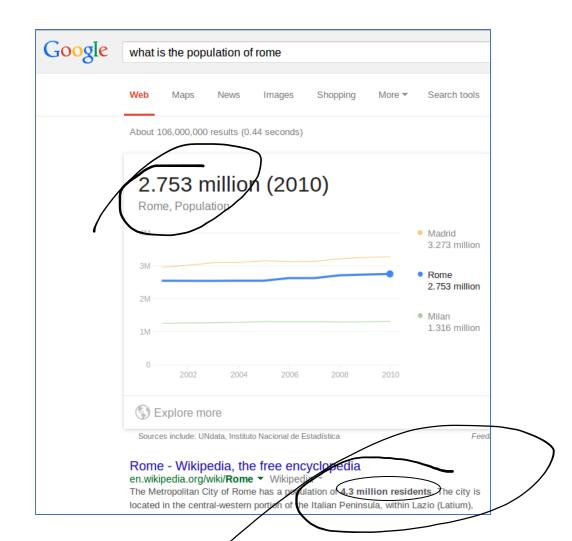
- Factoid QA needs information extraction, followed by ranking.
- Non-factoid QA needs filtering and summarization on top of that.
 - Query-based summarization
 - Focused summarization
- PERSON, LOCATION, ORGANIZATION and TIME questions are generally easier to handle than REASON and DESCRIPTION questions.

What role does IR play in QA?

- We need to **retrieve** and **rank** units of text in response to a query.
- The units should best be **indexed** beforehand.
- Snippets are very helpful.

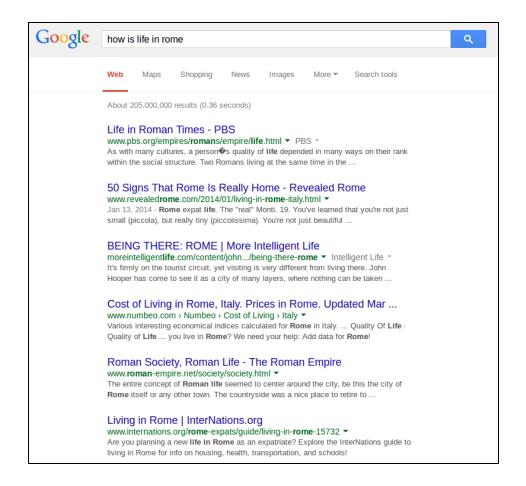
What role does IR play in QA?

• Snippets are very helpful.



What role does IR play in QA?

• Snippets are very helpful.



Question Answering from Text

- An idea originating from the IR community
- With massive collections of full-text documents, simply finding *relevant* documents is of limited use: we want *answers* from textbases
- QA: give the user a (short) answer to their question, perhaps supported by evidence.

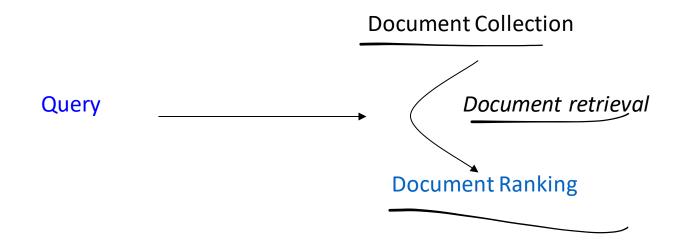
People Want to Ask Questions...

```
Examples from AltaVista query log
 who invented surf music?
 how to make stink bombs
 where are the snowdens of yesteryear?
 which english translation of the bible is used in official catholic
 liturgies?
 how to do clayart
 how to copy psx
 how tall is the sears tower?
 Examples from Excite query log (12/1999)
 how can i find someone in texas
 where can i find information on puritan religion?
Lwhat are the 7 wonders of the world
 how can i eliminate stress
 What vacuum cleaner does Consumers Guide recommend
 Around 12–15% of query logs
```

Applications

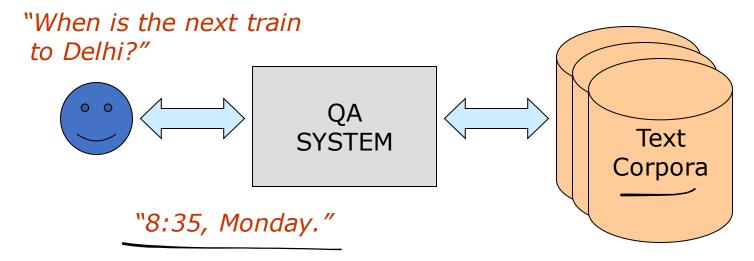
- General search for information
- Education
 - Tutoring systems
- Healthcare
- Customer service
- Travel reservations

Traditional IR



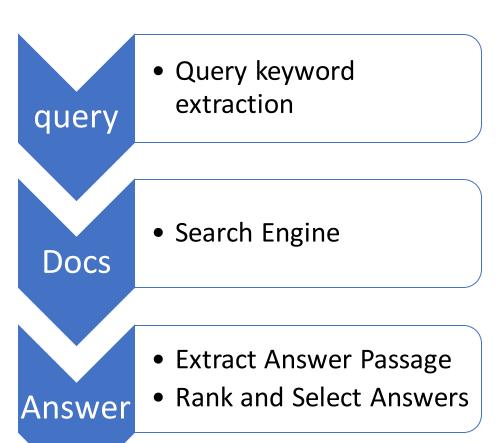
Question Answering

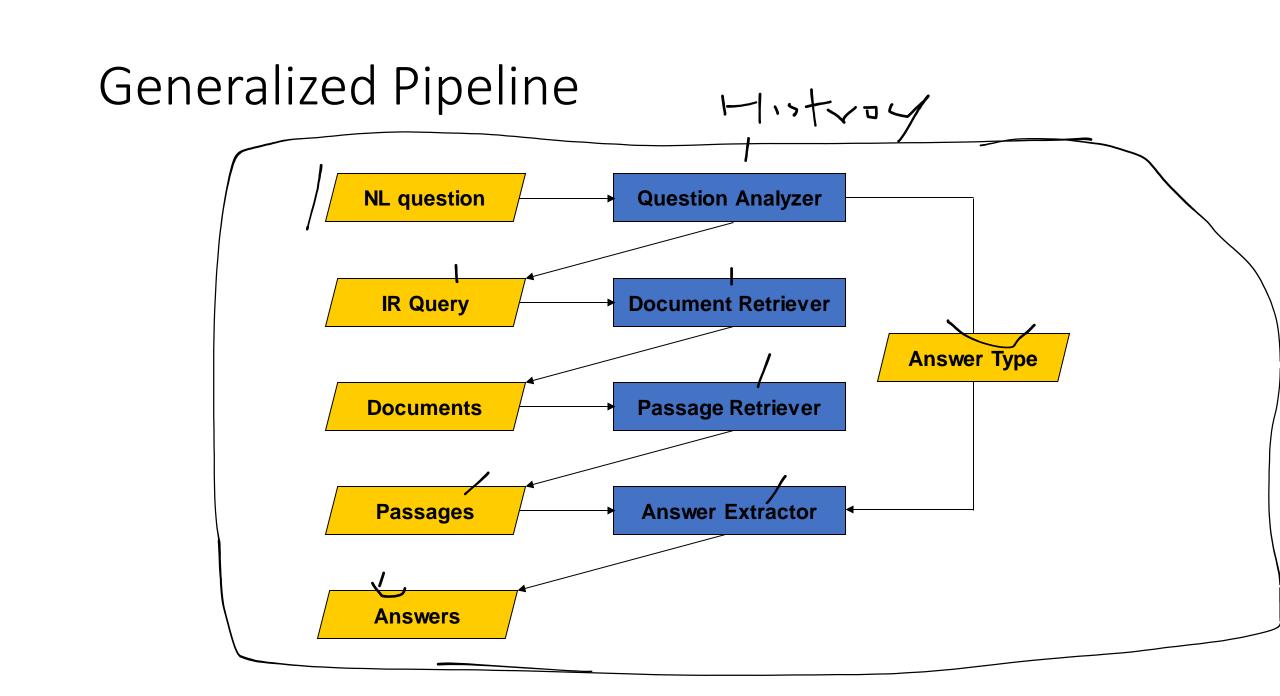
Typical pipeline



QA Pipeline

- Simple
 - Query keyword extraction
 - Send to search engine
 - Retrieve docs
 - Extract answer passage
 - Rank and select answer





Challenges

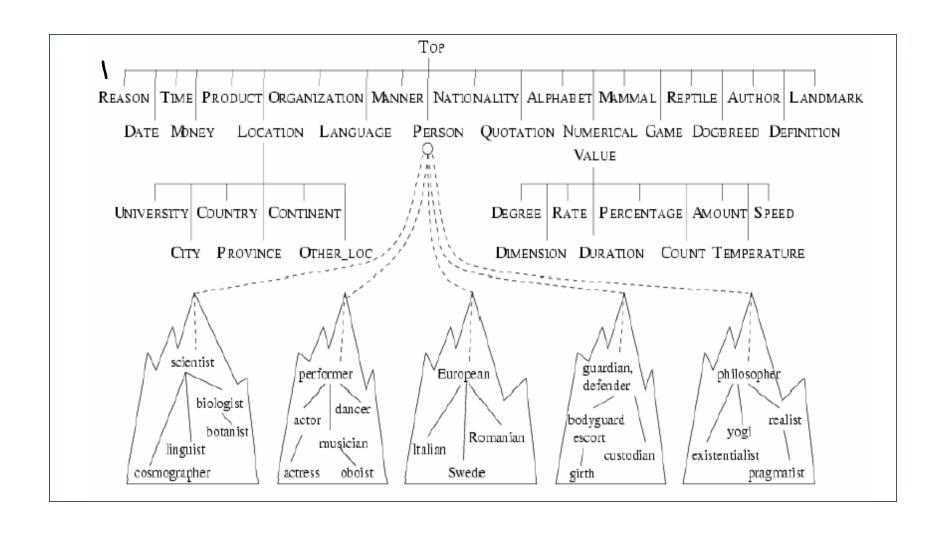
- Acquiring high-quality, high-coverage lexical resources
- Improving document retrieval ≤
- Improving document understanding
- Expanding to multi-lingual corpora
- Flexible control structure

 "beyond the pipeline"
- Answer Justification
 - Why should the user trust the answer?
 - Is there a better answer out there?

Question Analysis

- Question word cues
 - Who \rightarrow person, organization, location (e.g., city)
 - When \rightarrow date
 - Where → location
 - What/Why/How \rightarrow ??
- Head noun cues
 - What city, which country, what year...
 - Which astronaut, what blues band, ...
- Adjective cues
 - How long, how fast, how far, how old, ...

Answer Types



Advanced Issues in IR

- Query Expansion
 - Typical queries very short
 - Expand user query using an initial search and taking words from top N docs, using a thesaurus, using term clustering or WordNet to find synonyms....
- Tasks beyond Ad Hoc query support
 - Passage Retrieval, Multilingual IR, Speech IR, Summarization, Question Answering...

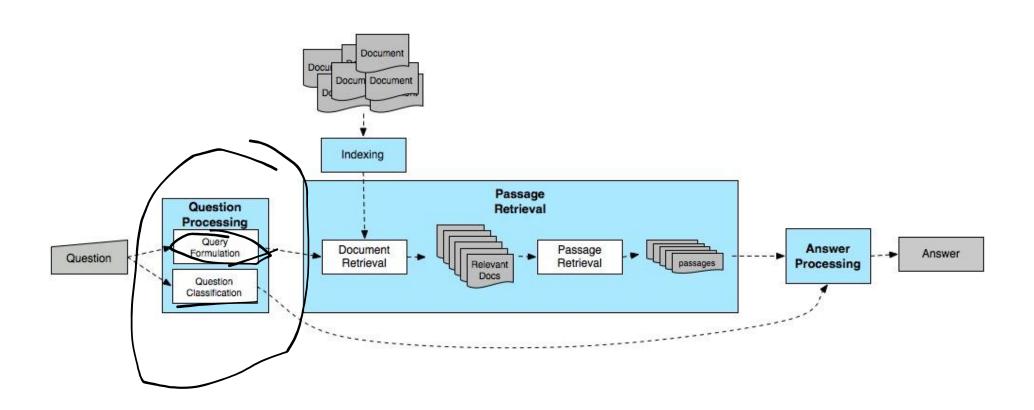
Question-Answering Systems

- Beyond retrieving relevant documents -- Do people want answers to particular questions?
- Three kinds of systems
 - Finding answers in document collections
 - Interfaces to relational databases
 - Mixed initiative dialog systems
- What kinds of questions do people want to ask?

Factoid Questions

Question	Answer
Where is the Louvre Museum located?	in Paris, France
What's the abbreviation for limited partnership?	L.P.
What are the names of Odin's ravens?	Huginn and Muninn
What currency is used in China?	the yuan
What kind of nuts are used in marzipan?	almonds
What instrument does Max Roach play?	drums
What's the official language of Algeria?	Arabic
What is the telephone number for the University of	(303)492-1411
Colorado, Boulder?	
How many pounds are there in a stone?	14

Typical Q/A Architecture



Question Processing

- Two main tasks
 - Question classification: Determine the type of the answer
 - Query formulation: Extract keywords from the question and formulate a query

Answer Types

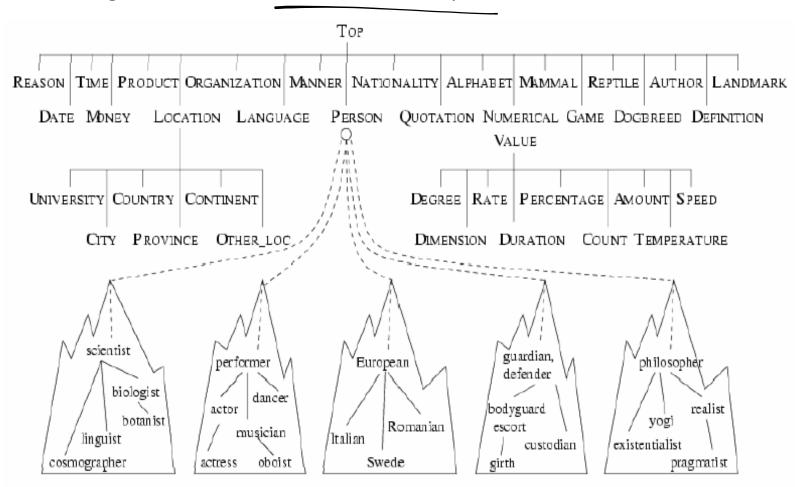
- Factoid questions...
 - Who, where, when, how many...
 - Answers fall into limited, fairly predictable set of categories
 - Who questions will be answered by...
 - Where questions will be answered by ...
 - Generally, systems select answer types from a set of Named Entities, augmented with other types that are relatively easy to extract

Answer Types Can Be More Complicated

- Who questions can have organizations or countries as answers
 - Who sells the most hybrid cars?
 - Who exports the most wheat?
- Which questions can have people as answers
 - Which president went to war with Mexico?

Taxonomy of Answer Types

- Contains ~9000 concepts reflecting expected answer types
- Merges NEs with the WordNet hierarchy



Answer Type Detection

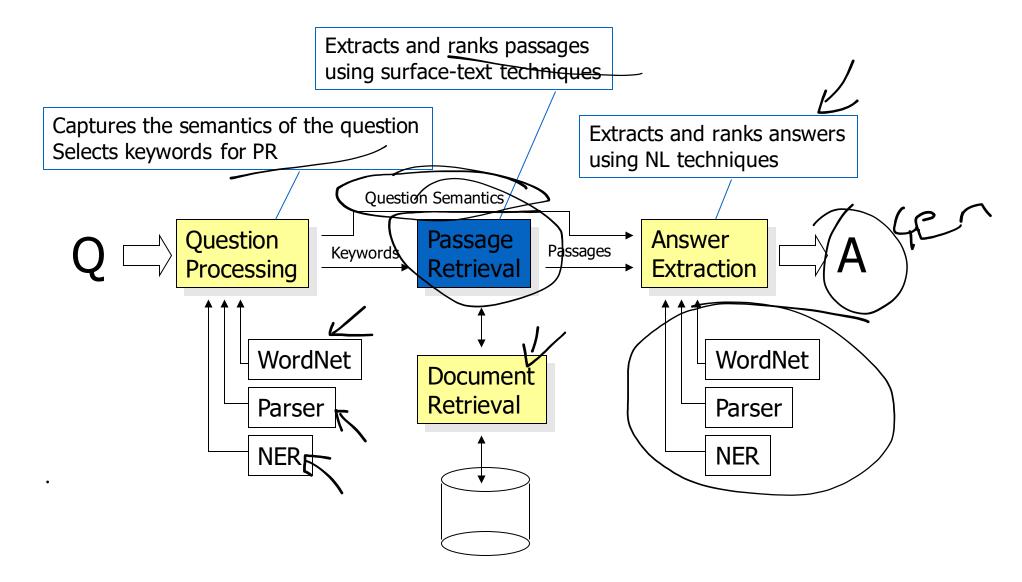
- Use combination of hand-crafted rules and supervised machine learning to determine the right answer type for a question
- But how do we make use of this answer type once we hypothesize it?

Query Formulation: Extract Terms from Query

- Questions approximated by sets of unrelated words (lexical terms)
- Similar to bag-of-word IR models

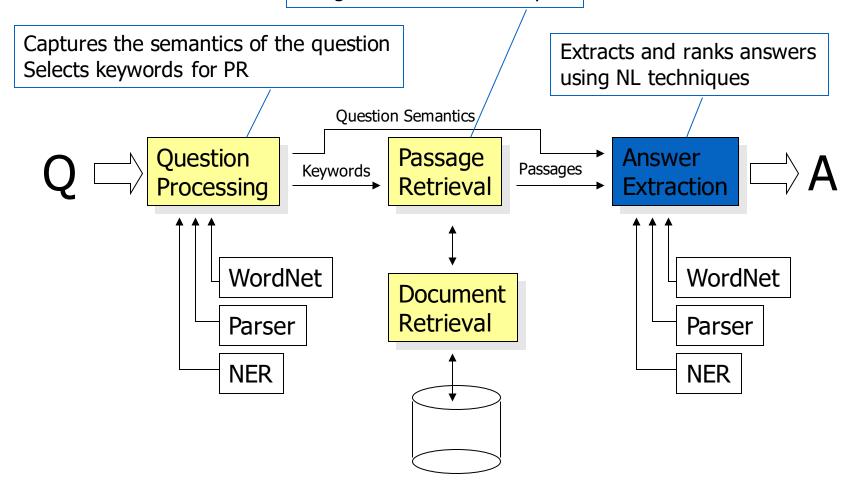
Question (from TREC QA track)	Lexical terms
Q002: What was the monetary value of the Nobel Peace Prize in 1989?	monetary, value, Nobel, Peace, Prize
Q003: What does the Peugeot company manufacture?	Peugeot, company, manufacture
Q004: How much did Mercury spend on advertising in 1993?	Mercury, spend, advertising, 1993
Q005: What is the name of the managing director of Apricot Computer?	name, managing, director, Apricot, Computer

Passage Retrieval



Answer Extraction

Extracts and ranks passages using surface-text techniques



Ranking Candidate Answers

Q066: Name the first private citizen to fly in space.

Answer type: Person

n Text passage:

"Among them was Christa McAuliffe, the first private citizen to fly in space. Karen Allen, best known for her starring role in "Raiders of the Lost Ark", plays McAuliffe. Brian Kerwin is featured as shuttle pilot Mike_Smith..."

Ranking Candidate Answers

Q066: Name the first private citizen to fly in space.

- Answer type: Person
- Text passage:

"Among them was Christa McAuliffe, the first private citizen to fly in space. Karen Allen, best known for her starring role in "Raiders of the Lost Ark", plays McAuliffe. Brian Kerwin is featured as shuttle pilot Mike_Smith..."

- Best candidate answer: Christa McAuliffe
- How is this determined?

Features Used in Answer Ranking

- Number of question terms matched in the answer passage
- Number of question terms matched in the same phrase as the candidate answer
- Number of question terms matched in the same sentence as the candidate answer
- Flag set to 1 if the candidate answer is followed by a punctuation sign
- Number of question terms matched, separated from the candidate answer by at most three
 words and one comma
- Number of terms occurring in the same order in the answer passage as in the question
- Average distance from candidate answer to question term matches

How does this approach compare to IE-based Q/A?

- When was Barack Obama born?
- Where was George Bush born?
- What college did John McCain attend?
- When did John F Kennedy die?

Is Q/A Different on the Web?

- In TREC (and most commercial applications), retrieval is performed against a small closed collection of texts
- More noise on the Web and more diversity
 - Different formats
 - Different genres
- How likely are you to find the actual question you asked?
- How likely are you to find a declarative version of your question?

QA over Semi-Structured data

- Intuitive ways of accessing RDF data become more and more important.
- Question answering approaches have been proposed as a good compromise between intuitiveness and expressivity.

general way: a triple-based representation

```
e.g. Who wrote The Neverending Story? (PowerAqua):
```

<[person,organization], wrote, Neverending Story>.

<Writer, IS_A, Person>

<Writer, author, The Neverending Story>

```
1. (a) Which cities have more than three universities?
  (b) <[cities], more than, universities three>
  (c) SELECT ?y WHERE {
          ?x rdf:type onto:University.
                                                     the original semantic structure
          ?x onto:city ?y.
                                                     of the question can not be
                                                     faithfully captured using triples.
          HAVING (COUNT(?x) > 3)
2. (a) Who produced the most films?
  (b)<[person,organization], produced, most films>
  (c) SELECT ?y WHERE {
          ?x rdf:type onto:Film.
          ?x onto:producer ?y.
```

ORDER BY DESC(COUNT(?x)) OFFSET 0 LIMIT 0

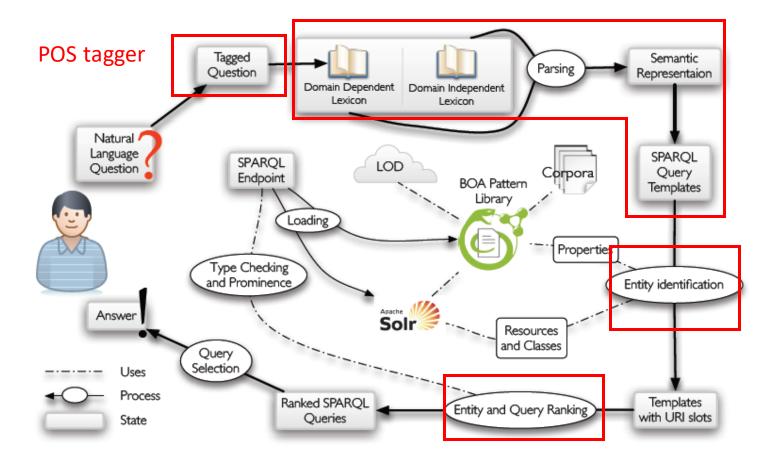


Figure 1: Overview of the template based SPARQL query generator.

Who produced the most films?

POS tagger

(a) who/WP produced/VBD the/DT most/JJS films/NNS

Parsing and template generation

• Domain independent lexicon:

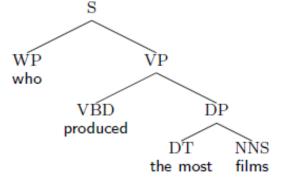
107 entries: light verbs, question words, determiners, negation words, coordination and the like.

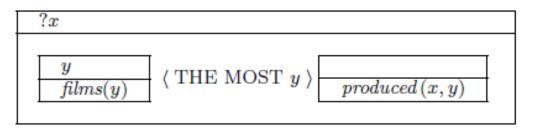
- (b) Covered tokens: who, the most, the, most
- Domain dependent lexicon: built on-the-fly.
- POS tag ——> syntactic and semantic properties.
- (c) <u>Building entries for: produced/VBD, films/NNS</u>

• POS tag ——> syntactic and semantic properties.

Heuristics:

- ✓ Named entities, resources.
- ✓ Nouns, classes, properties.
- ✓ Verbs, properties. If no contribution, instead by noun (Which cities have more than 2 million inhabitants?)





SPARQL templates:

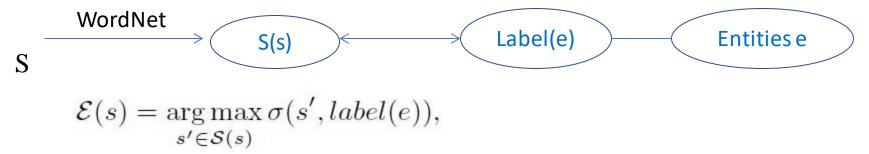
Who produced the most films?

```
(a) SELECT ?x WHERE {
       ?x ?p ?y.
       ?y rdf:type ?c.
   ORDER BY DESC(COUNT(?y)) LIMIT 1 OFFSET 0
   Slots:
       <?c, class, films>
       <?p, property, produced>
(b) SELECT ?x WHERE {
       ?x ?p ?y.
  ORDER BY DESC(COUNT(?y)) LIMIT 1 OFFSET 0
  Slots:
       <?p, property, films>
```

• Entity identification

String s—knowledge base K —similar entity

• Generic approach

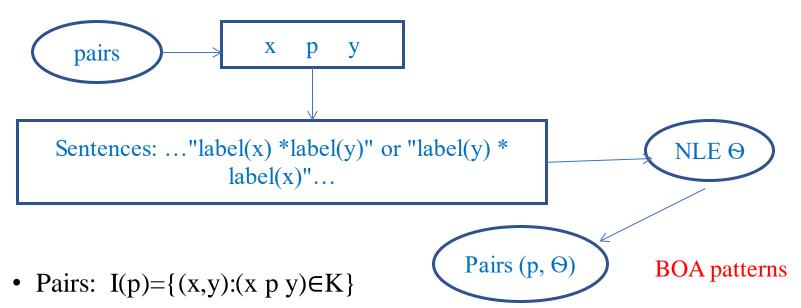


Property detection

Large number of expressions can be used to denote the same predicate.

(X, the creator of Y and Y is a book by X)

BOA pattern library



- NLE Θ : the form ?D? representation ?R? or ?R? representation ?D? **Distinguish patterns that are Specific to property p**.
- Support $sup(\theta, \mathbf{p}) = \log \left(\max_{(s, o) \in \mathcal{I}(\mathbf{p})} l(s, o, \theta, \mathbf{p}) \right) \log(|\mathcal{I}(\mathbf{p}, \theta)|).$
- Typicity $typ(\theta, \mathbf{p}) = \sum_{s \in S} \left(\frac{\delta(d(\mathbf{p}), d(\theta, s)) + \delta(r(\mathbf{p}), r(\theta, s))}{2|S|} \right) \log(|S| + 1),$
- Specificity $spec(\theta) = \log\left(\frac{|P|}{|M(\theta)|}\right),$

$$c(\theta, \mathbf{p}) = sup(\theta, \mathbf{p}) \cdot typ(\theta, \mathbf{p}) \cdot spec(\theta).$$

Query ranking and selection

• String similarity, prominence of entities and the schema of the knowledge base to score a query.

$$\varphi(e) = \begin{cases} \log_2 |\{(\mathtt{x},\mathtt{y}) : \mathtt{x} \ \mathtt{e} \ \mathtt{y}\}| & \text{if \mathtt{e} is a property} \\ \log_2 |\{(\mathtt{x},\mathtt{y}) : \mathtt{x} \ \mathtt{y} \ \mathtt{e}\}| & \text{else}, \end{cases}$$

$$score(e) = \alpha \max_{s' \in S(s)} \sigma(s', label(e)) + (1 - \alpha)\varphi(e),$$

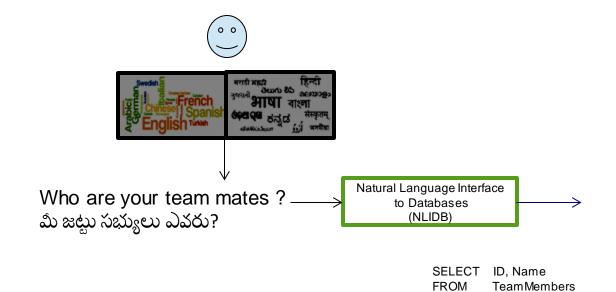
• Entities score:

• type checks on queries . (?x p e e p ?x)

Return:

the highest scored query with a non-empty result.

NLIDB Introduction



WHERE ID > 100

UB

Outline

- History
- Advantages
- Challenges
- Approaches to NLIDB
 - Syntactic
 - Semantic

History

- NLIDB has been around since late 1960s (and longer than relational database systems)
- NLIDB research peaked in 1980s, but failed to gain commercial acceptance—why?
 - Simpler interfaces took off: graphical and form-based
 - More precise interfaces became easier: SQL
 - Intrinsic problems with NL interfaces
- How would DBMS development affect NLIDB?
 - Object-oriented DBMS? More complex structures?
 - Pervasiveness pushing the envelope of usability?

Advantages

- User friendly
 - User doesn't need to learn artificial languages
- Better for some questions
 - Which research center has no programmers?
- Discourse
 - Some NLIDBs support elliptical expressions

Challenges

- Current NLIDBs deal with subset of NLs
 - Domain, FAQs patterns etc..
- Inappropriate medium -> Forms
- Linguistic problems
 - Modifier Attachment
- List all the students in the college with ID card
 - Quantifier Scoping
- Has every student taken some course
 - Conjunction disjunction
- List all students who live in Bakul and OBH
- What is the average score of MS and MTech students in NLP
- Anaphora , Elliptical expressions etc...

NLIDB Introduction

$$q_{NL} \longrightarrow q_{dbL}$$

Major focus in NLIDB

- Inherent Ambiguity in Natural languages
- Domain Expert dependency
- Mapping:
 - Semantic items (NL): Triples (dbL)

> Surface level models
> Syntax/Semantic level models

Surface Level: Keyword Matching Model

- Concentrates on Identifying Domain keywords
- Matches with the semantic patterns
- Apt for simple queries
- Can't handle complex queries
- Needs large number of patterns
- Tends to Language dependent
- Ex: PRECISE

Syntax/Semantic level Model

- Depends on linguistic information
 - Tokenizer, Morph Analysis, POS Tagger, Chunker
- Uses PST/Dependency trees
- Extracts domain terms
- Uses semantic frames