Processing of large document collections

Part 11 (Information extraction: multilingual IE, IE from web, IE from semi-structured data; Question answering systems)

Helena Ahonen-Myka

Spring 2005

Multilingual IE

- assume we have documents in two languages (English/French), and the user requires templates to be filled in one of the languages (English) from documents in either language
  - ”Gianluigi Ferrero a assisté à la réunion annuelle de Vercom Corp à Londres.”
  - ”Gianluigi Ferrero attended the annual meeting of Vercom Corp in London.”

Both texts should produce the same template fill:

- <meeting-event-01> :=
  - organisation: ’Vercom Corp’
  - location: ’London’
  - type: ’annual meeting’
  - present: <person-01>
- <person-01> :=
  - name: ’Gianluigi Ferrero’
  - organisation: UNCLEAR

Multilingual IE: three ways of addressing the problem

- 1. solution
  - a full French-English machine translation (MT) system translates all the French texts to English
  - an English IE system then processes both the translated and the English texts to extract English template structures
  - the solution requires a separate full IE system for each target language (here: for English) and a full MT system for each language pair

Multilingual IE: three ways of addressing the problem

- 2. solution
  - separate IE systems process the French and English texts, producing templates in the original source language
  - a ’mini’ French-English MT system then translates the lexical items occurring in the French templates
  - the solution requires a separate full IE system for each language and a mini-MT system for each language pair

Multilingual IE: three ways of addressing the problem

- 3. solution
  - a general IE system, with separate French and English front ends
  - the IE system uses a language-independent domain model (ontology) in which ’concepts’ are related via bi-directional mappings to lexical items in multiple language-specific lexicons
  - this domain model is used to produce a language-independent representation of the input text -> a discourse model

Multilingual IE: three ways of addressing the problem

- 3. solution continues…
  - the required information is extracted from the discourse model and the mappings from concepts to the English lexicon are used to produce templates with English lexical items
  - the solution requires a separate syntactic/semantic analyser for each language, and the construction of mappings between the domain model and a lexicon for each language

7

IE from web

- problem setting: data is extracted from a web site and transformed into structured format (database records, XML documents)
- the resulting structured data can then be used to build new applications without having to deal with unstructured data
  - e.g., price comparisons
- challenges:
  - thousands of changing heterogeneous sources
  - scalability: speed is important -> no complex processing possible

8

IE from web

- a **wrapper** is a piece of software that can translate an HTML document into a structured form (~database tuple)
- critical problem:
  - How to define a set of extraction rules that precisely define how to locate the information on the page?
- for any item to be extracted, one needs an extraction rule to locate both the beginning and end of the item
  - extraction rules should work for all of the pages in the source

9

Example: country codes

<HTML><TITLE>Some Country Codes</TITLE>

<BODY>

<B>Congo</B> <I>242</I>

<B>Egypt</B> <I>20</I>

<B>Belize</B> <I>501</I>

<B>Spain</B> <I>34</I>

<HR></BODY></HTML>

Extract: {<Congo, 242>, <Egypt, 20>, <Belize, 501>, <Spain, 34>}

10

Wrapper induction system

- input: a set of web pages labeled with examples of the data to be extracted
  - the user provides the initial set of labeled examples
  - the system can suggest additional pages for the user to label
- output: a set of extraction rules that describe how to locate the desired information on a web page

11

IE from semi-structured text

Capitol Hill - 1 br twnhme. Fplc D/W W/D. Undrgrnd Pkg incl $675. 3 BR, upper flr of turn of ctry HOME. incl gar, grt N. Hill loc $995. (206) 999-9999

  (This ad last ran on 08/03/97.)

 <hr>

12

IE from semi-structured text

- 2 templates extracted:
  - Rental:
    - Neighborhood: Capitol Hill
    - Bedrooms: 1
    - Price: 675
  - Rental:
    - Neighborhood: Capitol Hill
    - Bedrooms: 3
    - Price: 995

13

IE from semi-structured text

- the sample text (rental ad) is not grammatical nor has a rigid structure
  - we cannot use a natural language parser as we did before
  - simple rules that might work for structured text do not work here

14

Rule for neighborhood, number of bedrooms and associated price

- Pattern:: *(*Nghbr*) *(*Digit*) ' ' *Bdrm* * '$' (*Number*)
- Output:: Rental {Neighborhood $1} {Bedrooms $2} {Price $3}
  - assuming the semantic classes *Nghbr* (neighborhood names for the city) and *Bdrm*

15

Other trends in IE

- semi- and unsupervised methods
- ACE (Automatic Content Extraction) evaluations
  - extraction of general relations from text: person in a location; person has some social relation to another person, etc.
- cross-document processing
  - e.g. error correction, when the slot values for all templates are known
- backtracking in the process
  - now errors on the earlier levels propagate into later levels
  - could one backtrack and correct errors made earlier, and start then again?

16

Question answering (QA) systems

- Moldovan, Harabagiu, et al: The structure and performance of an open-domain question answering system, 2000
- Cooper, Rüger: A simple question answering system, 2000
- Aunimo, Makkonen, Kuuskoski: Cross-language question answering for Finnish, 2004

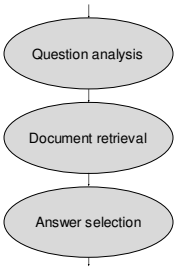
17

Problem setting

- a user gives a natural language question
  - "When did test pilot Chuck Yeager break the sonic barrier?"
- the question answering system returns an answer that can be
  - an exact answer (word, phrase…)
  - a snippet of text, in which the answer can be found

18

Architecture of a Q/A system



19

Question analysis

- question classification
  - predefined question classes, e.g. when, who, where, whom, why, description
- answer type
  - what kind of answer are we looking for?
- keyword extraction
  - which keywords should be given to the search engine?

20

Question classification

- often the question word tells the question class
  - who question -> who class
- what, which, how, and name are less clear
  - What **time** is the train arriving?
  - What **city** is the train arriving at?
  - What is the name of the **driver** of the train?
- question focus is found
  - a phrase in the question that disambiguates it and emphasizes the type of answer being expected
  - time -> when; city -> where; driver -> who

21

Answer type

- question word and question focus are used to decide the type of the answer
- *when, where, why* are straightforward:
  - when -> time
  - where -> place
  - why -> reason

22

Answer type

- who, whom
  - often the answer is a person's name
    - Who is the president of Finland?
  - can be a description
    - Who is Bill Gates?
    - pattern: (who|whom) (is|are|was) ProperNoun
  - can be a group of people
    - Who beat England in the relay? (USA and Canada)

23

Answer type

- what, which, name
  - answer type depends on the question focus
  - one solution:
    - if question focus not found -> answer-type = *name*
    - else if question focus describes a person -> answer-type = *person*
    - else answer-type = question focus
- how
  - how old -> *age*
  - how much -> *quantity*
  - how long -> *distance*
  - default: how -> *manner*

24

Answer type

- named entity recognition can be used
  - speed, temperature, money, place, city, country, person, year, time, length, reason, company, number, quoted, name

25

Keyword extraction

- query to the search engine:
  - named entities
  - if part-of-speech or syntactic analysis done:
    - nouns (or nouns+verbs) can be selected
  - keywords:
    - test pilot Chuck Yeager sonic barrier

26

Document retrieval

- the search engine returns documents (or paragraphs) which match the keywords
- challenge: balance between
  - getting enough documents to guarantee the presence of the answer
  - getting too many -> the answer selection phase slows down

27

Document retrieval

- first, a rigid query can be given
  - test AND pilot AND Chuck AND Yeager AND sonic AND barrier
- if the query does not return enough results, it is relaxed (keywords are dropped)
- additional conditions can be stated
  - the keywords have to occur within a paragraph (or within n paragraphs)

28

Document retrieval

- keywords:
  - test pilot Chuck Yeager sonic barrier
- document fragment is found:
  - "For many, seeing **Chuck Yeager** – who made his historic super**sonic** flight **Oct. 14, 1947** – was the highlight of this year's show, in which…"

29

Answer selection

- candidate answer extraction
  - mark up of regions that could be answers
- candidate scoring
  - heuristics are used to evaluate how likely a candidate is a correct answer
- candidate weighting
  - scores are combined into one final score

30

Candidate answer extraction

- retrieved text fragments (documents, a set of paragraphs) are split to sentences
  - (some of the) keywords occur in these fragments
- question class, answer type, and keywords guide candidate answer extraction
- named entity recognition (and other linguistic analysis) can be done first

31

Candidate answer extraction

- semantic knowledge can be used:
  - the question's answer type is looked up in WordNet and all of its hyponyms are found
    - example: answer type = "city"
    - WordNet: Helsinki (Tampere, Luanda…) is a kind of city
  - a regular expression is then built by taking a disjunction of those hyponyms
    - (Helsinki|Tampere|Luanda|…)
  - any region of text that matches the regular expression is marked up as a candidate answer

32

Candidate answer extraction

- exceptions: person, description, general cases
- if the answer concept is a person:
  - regular expression that matches proper names is used
  - WordNet has 300 hyponyms for person (e.g. consumer, coward, defender, guardian…)
- if the answer concept is a description
  - descriptions are hard to define in terms of what words make them up
  - when an entity is first introduced in a text, it is often followed by a comma and then a description
    - "Bill Gates, Head of Microsoft, said today…"

33

Candidate answer extraction

- WordNet does not (and cannot) cover all the possible answers, e.g. all the lengths
- for many answer types, a pattern for general cases is defined
  - company: a sequence of proper nouns ending in (Ltd|Plc|Co|and Son|…)
  - length: any number followed by a unit of length (miles, km, ft,…)

34

Candidate answer extraction

- In Aunimo et al, each question class has a set of answer patterns that are matched to retrieved documents
- a generic answer pattern is instantiated with query terms:
  - Chuck Yeager [^\.\?\!]+ ((Jan|Feb|Mar|Apr|Aug|Sep|Oct|Nov|Dec)\. [1-9]{1,2}, [1-9]{4})
  - the pattern "knows" which part is a possible answer (= answer candidate)
  - the candidate extracted from example: Oct. 14, 1947

35

Candidate scoring

- a variety of heuristics can be used to evaluate, how likely a candidate is a correct answer
- for instance,
  - *score_comma_3_word:* if a comma follows the candidate, then: how many of the 3 following words appear in the question
  - *score_punctuation* = 1, if a punctuation mark immediately follows the candidate, 0 otherwise
  - *score_same_sentence*: the number of questions words that are in the same sentence as the candidate

36

Candidate scoring

- *score_description_before*: if the answer concept is a description, then the number of words immediately preceding the candidate that are question words
- *score_description_in*: similar to score_description_before, but counts question words that appear in the candidate

- the scoring heuristics are independent and they can be applied in any order

37

Candidate ranking

- heuristic scores are combined into one final score by linear combination
- weights of the heuristic scores (for instance):
  - score_comma_3_word: 1.2
  - score_punctuation: 1.1
  - score_same_sentence: 1.0
  - score_description_before: 2.0
  - score_description_in: 1.0
- frequency of the candidate occurrences can also be used to strengthen the likelihood of that answer being correct

38

Evaluation of question answering systems: TREC (8-10)

- participants were given a large corpus of newspaper/newswire documents and a test set of questions (open domain)
- a restricted class of types for questions
- each question was guaranteed to have at least one document in the collection that explicitly answered it
- the answer was guaranteed to be no more than 50 characters long

39

Example questions from TREC-9

- How much folic acid should an expectant mother get daily?
- Who invented the paper clip?
- What university was Woodrow Wilson president of?
- Where is Rider College located?
- Name a film in which Jude Law acted.
- Where do lobsters like to live?

40

More complex questions

- What is epilepsy?
- What is an annuity?
- What is Wimbledon?
- Who is Jane Goodall?
- What is the Statue of Liberty made of?
- Why is the sun yellow?

41

TREC

- participants returned a ranked list of five [document-id, answer-string] pairs per question
- all processing was required to be strictly automatic
- part of the questions were syntactic variants of some original question

42

Variants of the same question

- What is the tallest mountain?
- What is the world's highest peak?
- What is the highest mountain in the world?
- Name the highest mountain.
- What is the name of the tallest mountain in the world?

43

Examples of answers

- What is a meerkat?
  - The meerkat, a type of mongoose, thrives in…
- What is the population of Bahamas?
  - Mr. Ingraham's charges of 'impropriety' are unlikely to excite the 245,000 people of the Bahamas
- Where do lobsters like to live?
  - The water is cooler, and lobsters prefer that

44

TREC

- Scoring
  - if the correct answer is found in the first pair, the question gets a score 1
  - if the correct answer is found in the k^{th} pair, the score is 1/k (max k = 5)
  - if the correct answer is not found, the score is 0
  - total score for a system: an average of the scores for the questions

45

Performance of Moldovan, Harabagiu et al (TREC)

	Percentage of questions in top 5	Score
Short answer	68,1%	55,5%
Long answer	77,7%	64,5%

46

Cross-language QA

- Example: Finnish questions & English answers
- question processing
  - question in Finnish
    - "Milloin koelentäjä Chuck Yeager rikkoi äänivallin?"
  - translation into English word by word -> query
- candidates are retrieved from a collection of documents in English
- answer is extracted from candidates: "Oct. 14, 1947"

47

Cross-language QA

- translation problems with Finnish
  - compound words
    - tuliaselaki: tuliase + laki
    - kulttuuripääkaupunki: kultuuri + pääkaupunki
  - vocabulary
    - immigrate (en) = tulla siirtolaisena
    - compare: immigrer (fr), immigrare (it), imigrar (pt), immigreren (du), immigrar (sp)

48

Cross-language QA

– ambiguity
  - words can be ambiguous
    – kerros: hiekkakerros, asuinkerros
  - translation adds ambiguity
    – layer (kerros, kerrostuma, peite, taivukas)
    – floor (lattia, istuntosali, kerros, tanssilattia, pohja)

49

Disambiguation

- solution:
  – select keywords from the question, e.g. all nouns and verbs
  – take for each keyword the first translation (in a dictionary) for each sense
    - e.g. koelentäjä: 1 translation; rikkoa: 20 translations; äänivalli: 2 translations
  – build all combinations and make a query with all of them (40 queries)
    - use combinations which return documents (13)
    - number of translations after disambiguation: koelentäjä: 1 translation; rikkoa: 6 translations; äänivalli: 2 translations

50

Performance of Aunimo et al. (CLEF04)

- Cross-Language Evaluation Forum (CLEF), QA Track
  – answer is not guaranteed to be found
  – the system has to tell how confident it is that the answer is correct
  – one answer only can be returned
  – the answer is a word or a phrase
- Aunimo et al has two systems: the performance varies 22-29%

51

More complex QA tasks

- each question may require information from more than one document
  – Name 10 countries that banned beef imports from Britain in the 1990s.
- follow-up questions
  – Which museum in Florence was damaged by a major bomb explosion in 1993?
  – On what day did this happen?

52

Question-answering in a closed domain

- above, the types of questions belonged to some closed class, but the topics did not belong to any specific domain (open-domain topics)
- in practice, a question-answering system may be particularly helpful in some closed well-known domain, like within some company

53

Question-answering in a closed domain

- special features in real-life systems
  – the questions can have any type, and they may have errors and spoken-language expressions
  – the same questions (variants) probably occur regularly -> extensive use of old questions
  – closed domain: extensive use of domain-knowledge feasible
    - ontologies, thesauri, inference rules

54