Question Answering

Mandar Mitra

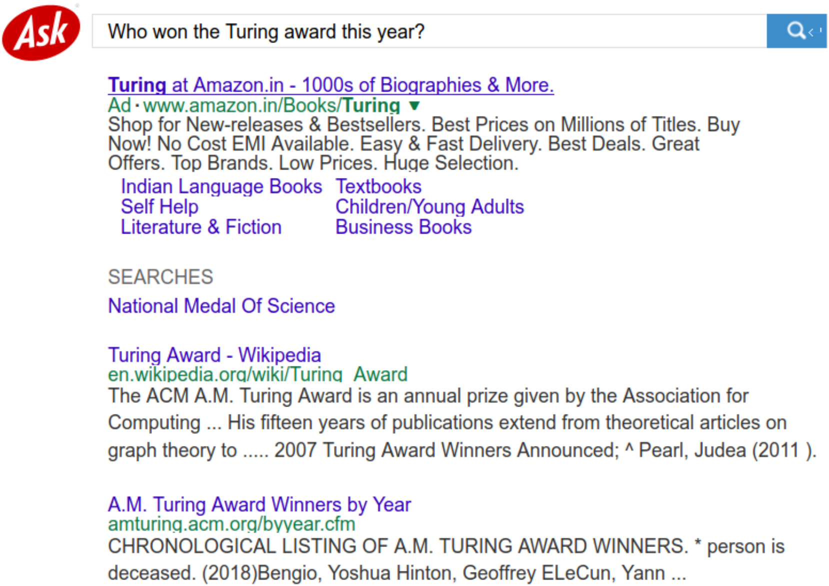
Indian Statistical Institute

Question answering

Motivation: given a question, system should provide an *answer* instead of requiring user to search for the answer in a set of documents

Example:
Q: *What is the height of Mount Everest?*
A: 8848 m.

Example systems (not really!)



Who won the Turing award this year?

Turing at Amazon.in - 1000s of Biographies & More.
Ad · www.amazon.in/Books/Turing ▾
Shop for New-releases & Bestsellers. Best Prices on Millions of Titles. Buy Now! No Cost EMI Available. Easy & Fast Delivery. Best Deals. Great Offers. Top Brands. Low Prices. Huge Selection.

Indian Language Books	Textbooks
Self Help	Children/Young Adults
Literature & Fiction	Business Books

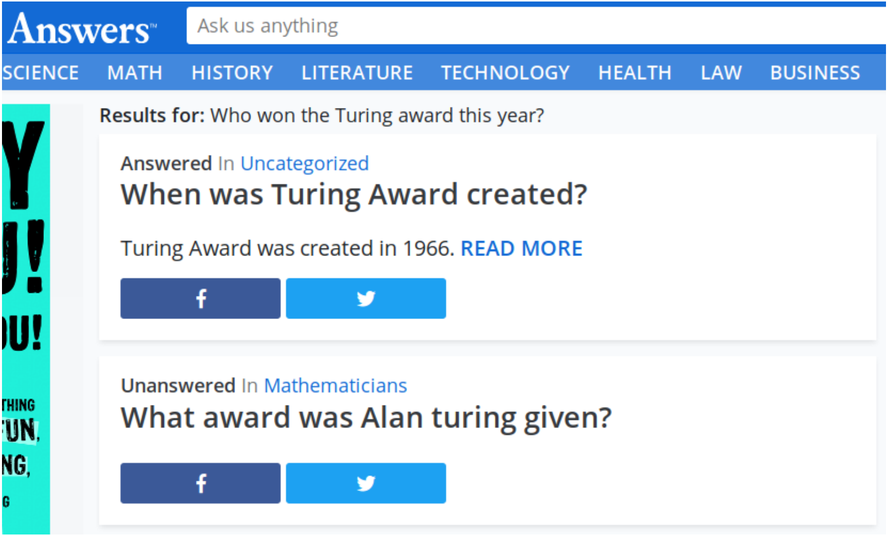
SEARCHES

National Medal Of Science

Turing Award - Wikipedia
en.wikipedia.org/wiki/Turing_Award
The ACM A.M. Turing Award is an annual prize given by the Association for Computing ... His fifteen years of publications extend from theoretical articles on graph theory to ..... 2007 Turing Award Winners Announced; ^ Pearl, Judea (2011).

A.M. Turing Award Winners by Year
amturing.acm.org/byyear.cfm
CHRONOLOGICAL LISTING OF A.M. TURING AWARD WINNERS. * person is deceased. (2018)Bengio, Yoshua Hinton, Geoffrey ELeCun, Yann ...

Example systems (not really!)



Results for: Who won the Turing award this year?

Answered In Uncategorized

When was Turing Award created?

Turing Award was created in 1966. READ MORE

Unanswered In Mathematicians

What award was Alan turing given?

Question taxonomy

- *Factoid* / trivia-type questions I: answer is often found in a text snippet from one or more documents (verbatim or simple morphological variation)
    - *wh* questions (*who*, *where*, *when*, etc.)
    - questions that may have yes/no answers
    - some *why* and *how* questions (see below)
    - *what* questions are hard
    - questions may be phrased as requests or commands

Question taxonomy

- *Factoid* / trivia-type questions I: answer is often found in a text snippet from one or more documents (verbatim or simple morphological variation)
    - *wh* questions (*who*, *where*, *when*, etc.)
    - questions that may have yes/no answers
    - some *why* and *how* questions (see below)
    - *what* questions are hard
    - questions may be phrased as requests or commands
- *Factoid* questions II: questions requiring simple reasoning / world knowledge to relate the question with the answer
    - *why*, *how* questions
      e.g. *How did Socrates die?* ⟶ (by) drinking poisoned wine
      How far is X from Y?

Question taxonomy

- *List* questions (non-factoid): fusion of partial answers scattered over several documents is necessary
  - *List 3 major rice producing nations.*
  - *What movies did AB appear in?*
  - *How do I assemble a bicycle?*

Question taxonomy

- *List* questions (non-factoid): fusion of partial answers scattered over several documents is necessary
    - *List 3 major rice producing nations.*
    - *What movies did AB appear in?*
    - *How do I assemble a bicycle?*
- Definition questions (non-factoid)
    - *Who was Picasso?*
    - *What are the Valdez Principles?*
    - *What does a defibrillator do?*

Question taxonomy

- *List* questions (non-factoid): fusion of partial answers scattered over several documents is necessary
  - *List 3 major rice producing nations.*
  - *What movies did AB appear in?*
  - *How do I assemble a bicycle?*
- Definition questions (non-factoid)
  - *Who was Picasso?*
  - *What are the Valdez Principles?*
  - *What does a defibrillator do?*
- Relationship, Superlative, . . .
- Reading comprehension, multiple-choice, fill in the blanks, . . .

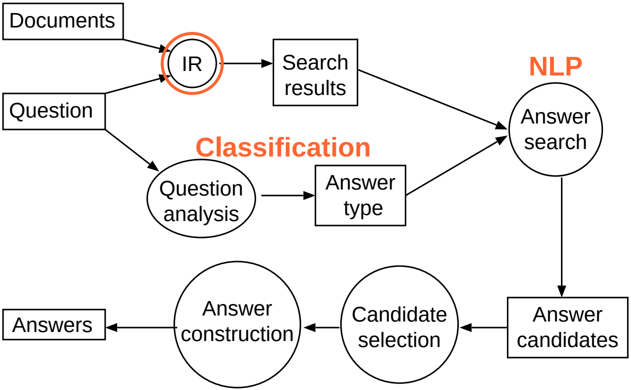
some pointers at the end

Question taxonomy

- Context questions / question series / sessions: questions have to be answered in the context of previous interactions with the user
  - *Who assassinated Indira Gandhi? When did this happen?*
  - ENTITY: Chandrayaan 2
    FACTOID: *When was it launched?*
    FACTOID: *When is it supposed to enter orbit?*
    LIST: *What are some other satellites launched by India?*
    OTHER: *Other*

return *nuggets* (relevant facts) about the target that have not already been provided

Traditional QA system architecture



General approach

1. **Question analysis:** find type of object that answers question: “when” - time, date “who” - person, organisation, etc.
2. **Document collection preprocessing:** prepare documents for real-time query processing
3. **Document retrieval (IR):** using (augmented) question, retrieve set of possible relevant documents/passages using IR
4. **Document processing (IE):** search documents for entities of the desired type and in appropriate relations using NLP
5. **Answer extraction and ranking:** extract and rank candidate answers from the documents
6. **Answer construction:** provide (links to) context, evidence, etc.

Question analysis - I

Keyword selection

- Objectives
  - document retrieval
  - locating candidate answer-bearing sentences / passages
- High-level approach: identify *question topic* , *question focus* , other 'supporting' words
  Examples:
  - In what city is NSCB International Airport located?
  - What is the population of West Bengal ?
  - What is the colour of *kurchi* flowers?
- General strategy: formulate very specific query → successive relaxation

Question analysis - II

Answer type identification: identify semantic type of the entity sought by the question

- Depends on the domain
- Derived from training data, e.g., logs
- Common types: PERSON, ORGANISATION, DATE, QUANTITY, PLACE, etc.
- *when*, *where*, *who*, *how many* — relatively easy to handle
- *which*, *what* — ambiguous
  - e.g. *What was the Beatles' first hit single?*
- Need *default* type and processing strategy

Question analysis - III

Additional constraints (on the answer entity)

- Relations (syntactic/semantic) that should hold between a candidate answer entity and other entities mentioned in the question

Document (pre)processing

- Preprocessing: detailed analysis of all texts in the corpus may be done a priori
    - e.g., annotate + index terms with semantic tags

Document (pre)processing

- Preprocessing: detailed analysis of all texts in the corpus may be done a priori
  - e.g., annotate + index terms with semantic tags
- Retrieval: initial set of candidate answer-bearing documents are selected from a large collection
  - boolean retrieval methods may be used profitably
  - passage retrieval may be more appropriate

Document (pre)processing

- Preprocessing: detailed analysis of all texts in the corpus may be done a priori
  - e.g., annotate + index terms with semantic tags
- Retrieval: initial set of candidate answer-bearing documents are selected from a large collection
  - boolean retrieval methods may be used profitably
  - passage retrieval may be more appropriate
- Analysis:
  - part of speech (POS) tagging
  - named entity identification (NEI): recognizes multi-word strings as names of companies/persons, locations/addresses, quantities, etc.
    - may use *authority lists* (table lookup), e.g., WordNet, CIA World Factbook
  - shallow/deep syntactic analysis: obtains information about syntactic relations, semantic roles

$\boxed{\text{Answer type identification} \quad \Longleftrightarrow \quad \text{NEI}}$

Answer extraction / ranking

- Look for strings whose semantic type matches that of the expected answer
  - matching may include subsumption
- Check additional constraints
  - select a window around matching candidate and calculate word overlap between window and query; OR
  - check how many distinct question keywords are found in a matching sentence, order of occurrence, etc.
- Check syntactic/semantic role of matching candidate
  - e.g. *Who killed Lee Harvey Oswald?*

    $\text{Ruby killed } \underbrace{\text{Oswald}}_{\text{OBJ}} \; \ldots \; \underbrace{\text{Oswald}}_{\text{SUBJ}} \text{ killed Kennedy.}$

Heavy NLP based systems

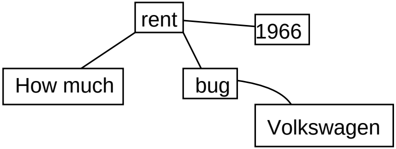
Falcon: methodology I

How much could you rent a Volkswagen bug for in 1966?

1. Question word normalization:
   - individual question words are spell-checked; spelling variants (if any) are included
     e.g., *Volkswangen* ↔ *Volkswagen*
   - questions are rephrased (if necessary) into a normalized form where the wh-word appears at the beginning

Falcon: methodology II

2. Question parsing: question is transformed into internal representation that captures question concepts (phrase heads) and binary dependencies between the concepts



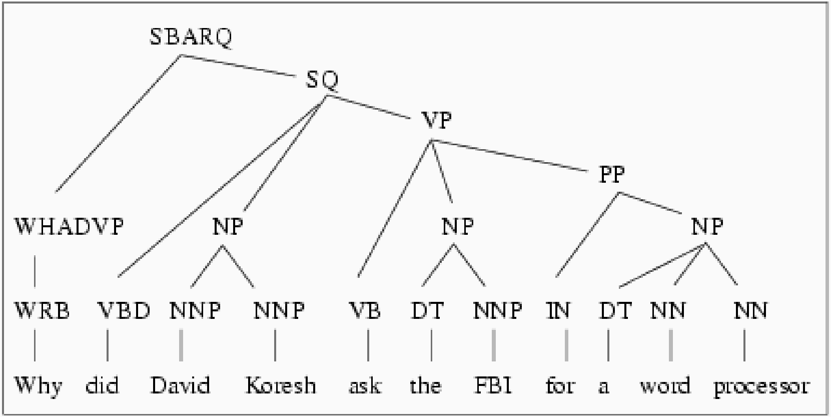
   (a) Parse the question (Collins, 1996).

   (b) Label all leaves of the parse tree as *skipnodes* or *non-skipnodes* (nouns, non-aux verbs, adjectives, adverbs).

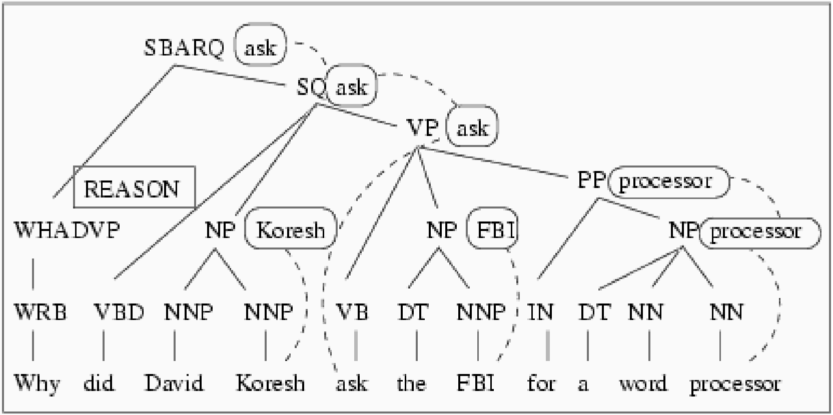
   (c) Propagate labels bottom-up. If there are multiple non-skipnode children:

      - the syntactic category of the parent selects one child node;
      - winning node is linked to all other non-skipnode siblings.

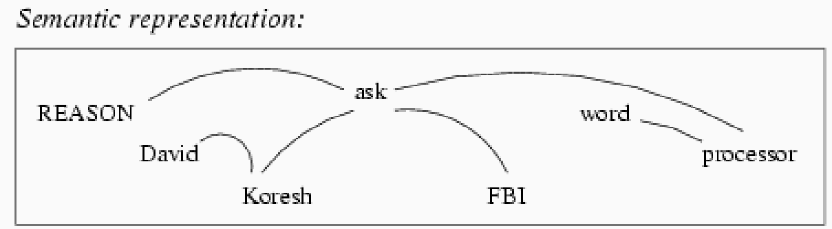
Falcon: methodology III



Falcon: methodology IV



Falcon: methodology V



Semantic representation:

Falcon: methodology VI

3. Answer type detection: semantic category of the expected answers determined based on:
   - question stem (*what*, *who*, *how much*) and its modifier (information obtained in Step 2)
   - WordNet-based answer type hierarchy

$$\boxed{\textit{How much} + \textit{rent} \Rightarrow \text{MONEY}}$$

Falcon: methodology VII

4. Keyword selection: subset of the question concepts are selected as keywords
   - part-of-speech information used

   Volkswagen, *bug*, *rent*

5. Keyword expansion: selected keywords are expanded with morphological, lexical, or semantic variants
   - morphological: *invented* → *inventor*
   - lexical: *kill* → *assassin*, *far* → *distance*
   - semantic: *like* → *prefer*

   rent → *rented*

Falcon: methodology VIII

6. Passage retrieval:
   - Boolean query formed from selected keywords
   - retrieval engine selects documents using Boolean query

     Volkswagen AND *bug*

   - engine selects smaller passages where all keywords are located in close proximity of one another
   - some additional text included as surrounding context
7. Passage filtering: passages not satisfying semantic constraints specified in the question are discarded

   in 1966

Falcon: methodology IX

8. Candidate answer identification: look for strings that have the same semantic type as the expected answer
   - named entity identifier may be used

     $1, *$29.95*
   - patterns may be used
     - e.g. for definition (*what is*) questions:
       $\langle AP \rangle$ *such as* $\langle QP \rangle$
       $\langle QP \rangle$ *is an* $\langle AP \rangle$
       $\langle AP \rangle$ (*also called* $\langle QP \rangle$)
9. Answer ranking: each candidate answer receives a relevance score according to lexical and proximity features such as distance between keywords
   - # of question words matched in the same phrase/sentence as the concept of expected answer type
   - average distance from candidate answer to a question word match

Falcon: methodology X

10. Answer selection: candidate answers with the highest relevance scores are selected

Impact of NLP/Resources

No NLP$^{(*)}$	0.028
No step 8	0.150
All steps	0.468
No WordNet	40% drop
No NER	70% drop

$(*)$ — no attempt is made to estimate the location of the relevant text fragments within retrieved passages

WordNet

- useful for deriving the answer type and keyword expansion
- especially important for hard questions (e.g. *what*) (precision drops by $>$ 60%)

Template / pattern matching based methods

InsightSoft: patterns

Definition: sequences or combinations of certain strings (letters, punctuation, space, special symbols (&, %, $, etc.), digits) and words/phrases or word classes

Example: *When was Mozart born?*

Answer patterns:

- In strict order: capitalized word; parenthesis; four digits; dash; four digits; parenthesis **(850)**
- In any order: capitalized word; “in”; four digits; “born” **(825)**
- . . .

InsightSoft: pattern construction

- Text corpus was studied systematically to identify expressions that can be interpreted as answers to questions of a definite type

  Expression: "Milan, Italy" (independent of context)
  Answers: *Where is Milan?*
  Pattern: city name; comma; country name

- Expressions may unintentionally convey information that can be interpreted as answering a certain question
- Pattern types:
  - ordered
  - unordered combination of pattern elements
    - 51 such lists constructed for TREC 2001
    - for each question type, 5–15 lists were used to find potential answers

InsightSoft: pattern scoring

- Score of a pattern:
  - high score — match strongly indicates that the right answer has been found
  - low score — may be present in a number of candidate answer strings both correct and wrong
    - text of candidate answers (and surrounding context) needs to be checked for additional indicators of validity e.g. presence of a specific type of word
- Score independent of actual question being answered
- Validity of a pattern for a given question type (and its score) can be tested on large text corpora

"Definition" Patterns

1. $\langle\ A$; is/are; [a/an/the]; $X\ \rangle$
   $\langle\ X$; is/are; [a/an/the]; $A\ \rangle$
   Example: Michigan's state flower is the apple blossom.
2. $\langle\ A$; comma; [a/an/the]; X; [comma/period] $\rangle$
   $\langle\ X$; comma; [a/an/the]; A; [comma/period] $\rangle$
   Example: Moulin Rouge, a cabaret
3. $\langle\ A$; [comma]; or; X; [comma] $\rangle$
   Example: shaman, or tribal magician,
4. $\langle\ A$; [comma]; [also] called; X [comma] $\rangle$
   $\langle\ X$; [comma]; [also] called; A [comma] $\rangle$
   $\langle\ X$; is called; $A\ \rangle$
   $\langle\ A$; is called; $X\ \rangle$
   Example: naturally occurring gas called methane

"Definition" Patterns

5. $\langle X$; dash; A; [dash] $\rangle$
   $\langle A$; dash; X; [dash] $\rangle$
   Example: nepotism - hiring relatives for the better jobs
6. $\langle X$; parenthesis; A; parenthesis $\rangle$
   $\langle A$; parenthesis; X; parenthesis $\rangle$
   Example: myopia (nearsightedness).

- Above patterns used for answering "*what is*" questions
- Also useful for "*who*", "*where*", etc.

InsightSoft: methodology

1. **Question analysis:**
   - identify *primary*, *secondary* and other (*marker*) query terms (based on *idf*)
   - identify question type
2. **Passage retrieval:**
   - use primary query words
   - for certain question categories, secondary searching terms are also used
     - e.g. *Who is (the) X of (the) Y?*
       where *X* is a post, and *Y* is the name of a country, company, etc.
   - for some question types, secondary query terms should be supplemented with related words
     - passage-based blind feedback
     - knowledge bases (WordNet, word lists, etc.)

InsightSoft: methodology

3. **Snippet construction:** retrieved passages are cut into 50-byte snippets (around the query words)
4. **Snippet analysis:**
   - snippets are analyzed to identify patterns
   - based on matching patterns, snippets are scored and ranked
   - if candidate answers do not match any pattern, they are scored based on lexical similarity between question and snippet

InsightSoft: assumptions

- Detailed categorization of question types
  - e.g. nine types of *Who*-type questions are distinguished (*Who-Post*, *Who-Author*, etc.)
- Diversity of patterns for each type
  - e.g. 23 patterns defined for *Who-Author*-type questions
- Recall of sufficiently large number of passages containing candidate answers

References

- *Natural language question answering: the view from here.* L. Hirschman, R. Gaizauskas. **Natural Language Engineering**, 7(4), December 2001.
- *Open-Domain Question-Answering.* John Prager. Foundations and Trends in Information Retrieval, 1(2), 2006. ★★★
- Misc. links, code repositories
  - `http://nlpprogress.com/english`
  - `https://allenai.org/aristo/`

Datasets

- *Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge.* Clark et al., Allen Institute for Artificial Intelligence. CoRR [cs.AI] 1803.05457v1, 2018.
- *SQuAD: 100,000+ Questions for Machine Comprehension of Text.* Rajpurkar et al., Stanford University. EMNLP 2016.
    - `https://rajpurkar.github.io/SQuAD-explorer/`
    - `https://www.kaggle.com/stanfordu/stanford-question-answering-dataset/`
- *WIKI QA: A Challenge Dataset for Open-Domain Question Answering.* Yang et al. EMNLP 2015.
- *MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text.* Richardson et al., Microsoft Research. EMNLP 2013.