NUS | Computing
National University of Singapore

CS4248: Natural Language Processing

Lecture 11 — Classification Applications

Announcements

- Project
  - Deadline for report: Thursday, Apr 18, 11.59 pm (+ launch of 2nd TEAMMATES session)
  - Deadline for 2nd TEAMMATES session: Thu, Apr 25, 11.59 pm
- Final Exam
  - Time/date: Monday, Apr 29, 17:00-19:00
  - Venue: MPSH1-B
  - Setup: Examplify (non-secure block internet) ➜ open-book exam
  - Survey: Loaner Laptops

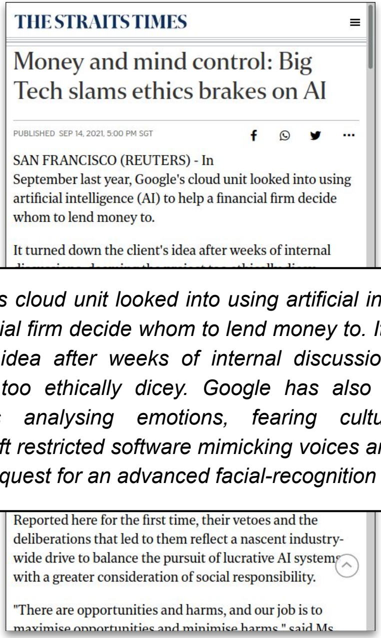
Outline

- **Text Summarization**
  - **Overview & Categorization**
  - Basic Architecture
  - Evaluation
  - Query-Focused Summarization
- Question Answering
  - Overview & Categorization
  - Factoid QA (Basic Architecture)
  - Core Components
  - Extended Concepts

Text Summarization

- Text Summarization — basic goal
  - Generate a condensed version of a (large) document or multiple documents
  - Summarization should convey the main idea of the original document(s) to the reader

- Wide range of applications
  - Outlines or abstracts of any document, article, etc.
  - Summaries of email threads
  - Action items from a meeting
  - Simplifying text by compressing sentences


THE STRAITS TIMES

Money and mind control: Big Tech slams ethics brakes on AI

PUBLISHED SEP 14, 2021, 5:00 PM SGT

SAN FRANCISCO (REUTERS) - In September last year, Google's cloud unit looked into using artificial intelligence (AI) to help a financial firm decide whom to lend money to.

It turned down the client's idea after weeks of internal

Reported here for the first time, their vetoes and the deliberations that led to them reflect a nascent industry-wide drive to balance the pursuit of lucrative AI systems with a greater consideration of social responsibility.

"There are opportunities and harms, and our job is to

Google's cloud unit looked into using artificial intelligence to help a financial firm decide whom to lend money to. It turned down the client's idea after weeks of internal discussions, deeming the project too ethically dicey. Google has also blocked new AI features analysing emotions, fearing cultural insensitivity. Microsoft restricted software mimicking voices and IBM rejected a client request for an advanced facial-recognition system.

Text Summarization — Dimensions

Input / Source

Single Document

- Input: single document
  (e.g., news article, web page, blog post, etc.)
- Common outputs:
  - abstract
  - outline
  - headline

Multiple Documents

- Input: group/set of documents
- Case 1: documents are about similar topic
  (e.g., multiple news stories about the same event)

  ➜ **Output: "proper" text summary**

- Case 2: documents are about diverse topics
  (e.g., all news stories over the course of a day)

  ➜ **Output: clusters / categories of document**
  (potentially with a text summary for each cluster/category)

Text Summarization — Dimensions

Trigger

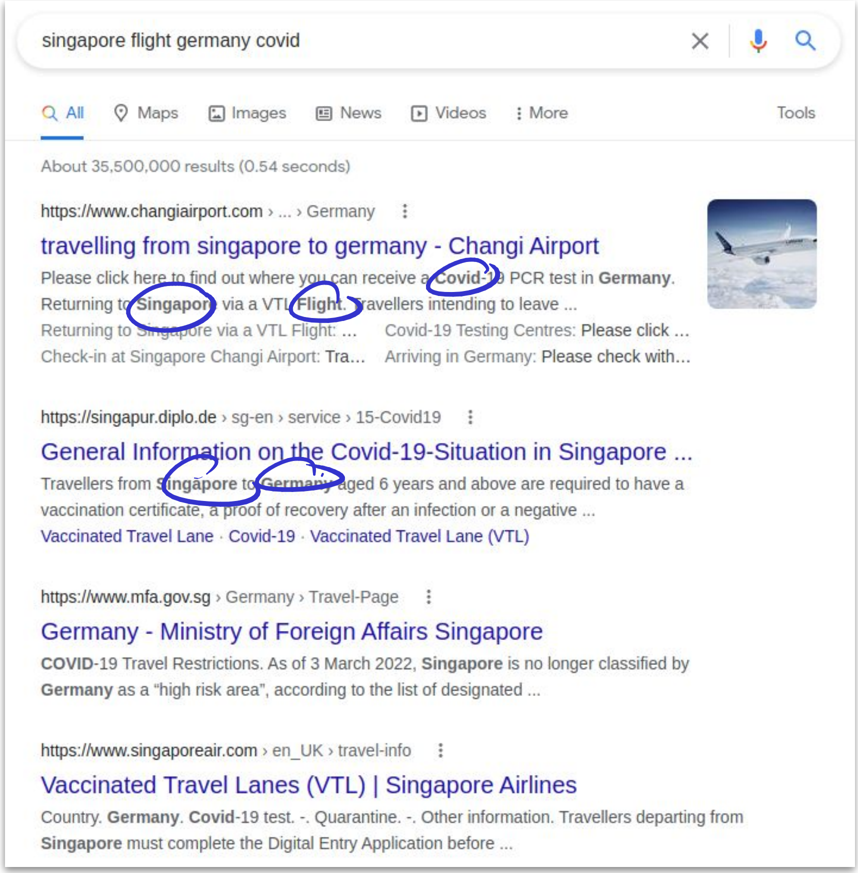
Generic Summarization

- "Just" summarize the content
- No additional factor/requirement/etc. driving the summarization process

Query-focused Summarization

- Summarize a document with respect to an information need expressed in a **user query**
- Kind of a complex Q&A task

Query-Focused Summarization — Example



Online search

- Summary = sentence snippets from the search result page
- Heuristics — pick snippets that
  - Include many search terms
  - Appear early in the document
  - Have special markups (e.g., bold)
  - …

Text Summarization — Dimensions

Summarization Approach

Extractive Summarization

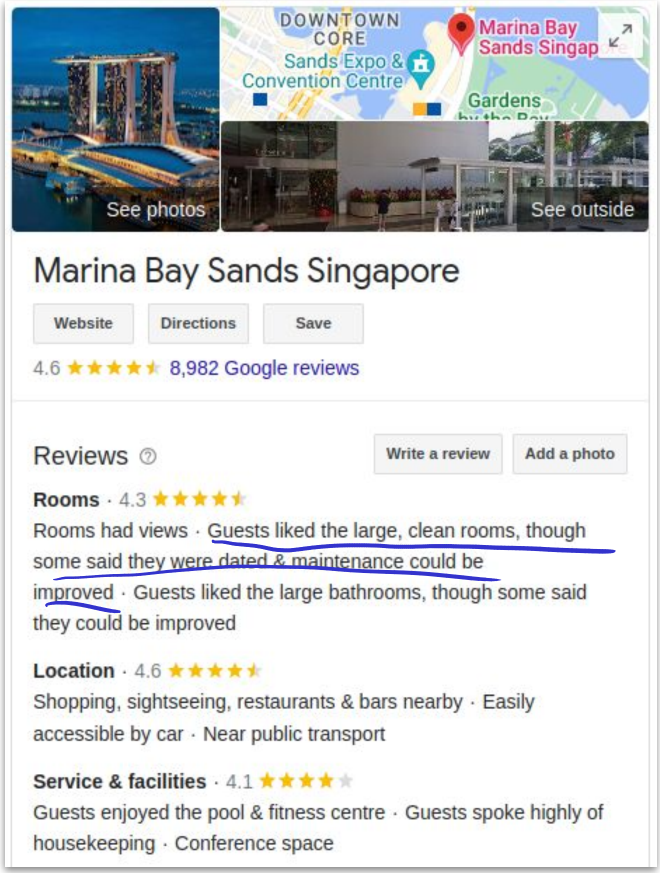
- Summary = selected phrases or sentences from source document(s)
- No "true" text generation task
- Challenge: risk of incoherent summaries

Abstractive Summarization

- Summary = newly generated text (potentially using completely different words)
- Advantage: generally much more coherent
- Challenge: generally more difficult (compared to extractive summarization)

Note: Both approaches can be combined, e.g: use extractive summarization to find subset of important sentences and that apply abstractive summarization over this subset.

Abstractive Summarization — Example



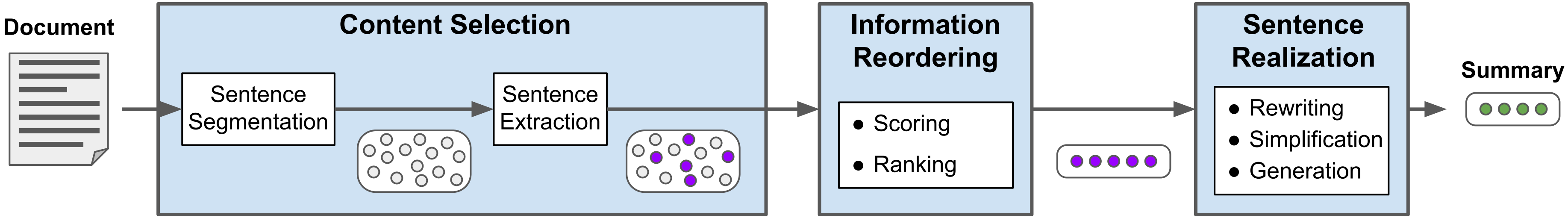
Google hotel review summary

- Identification of frequent phrases
  (with either positive or negative sentiment)
- Display of most common phrases
  (potentially a canonical version of similar phrases)
- Generation of very simple sentences
  (e.g.: "Guest liked […] but some said [...]")
- Sentence generation based on templates
  (disclaimer: my personal opinion; might be wrong!)
- Advantages
  - Simple but still appropriate results
  - "Safe" results (no risk of weird reviews)

Outline

- **Text Summarization**
  - Overview & Categorization
  - **Basic Architecture**
  - Evaluation
  - Query-Focused Summarization
- Question Answering
  - Overview & Categorization
  - Factoid QA (Basic Architecture)
  - Core Components
  - Extended Concepts

Summarization — 3 Core Stages



(1) Content Selection

- Choose sentence (or phrases) to extract

(2) Information Reordering

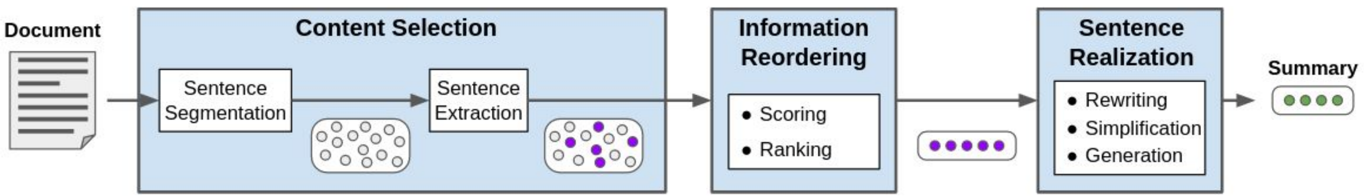
- Choose and order to place them

(3) Sentence Realization

- Clean up sentences; finalize summary

In-Lecture Activity (5-7 mins)

- Task: Come up with your own simple baseline for all 3 subtasks
  - Post your solution to Canvas > Discussions
    (individually or as a group; include all group members' names in the post)

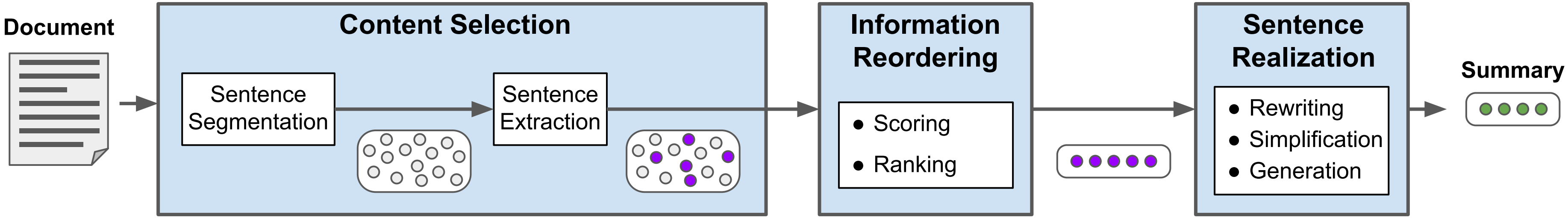


(1) Content Selection
- Choose sentence (or phrases) to extract

(2) Information Reordering
- Choose and order to place them

(3) Sentence Realization
- Clean up sentences; finalize summary

Summarization — (Most) Basic Algorithm



(1) Content Selection
- Choose sentence (or phrases) to extract

→ **How to do this part?**

(2) Information Reordering
- Choose and order to place them

→ Preserve original order
(easy; single document – but multiple documents?)

(3) Sentence Realization
- Clean up sentences; finalize summary

→ Extractive: Keep original sentence
(i.e., no rewriting, simplification, generation)

Content Selection — Baseline Algorithm

- Naive approach: Pick the first sentence(s)

Not logged in Talk Contributions Create account Log in

Article Talk Read View source View history Search Wikipedia

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Current events
Random article
About Wikipedia
Contact us
Donate

Contribute
Help
Learn to edit
Community portal
Recent changes
Upload file

Tools
What links here
Related changes
Special pages
Permanent link
Page information
Cite this page
Wikidata item

Print/export
Download as PDF
Printable version

In other projects
Wikimedia Commons

Singapore

From Wikipedia, the free encyclopedia

Coordinates: 1°17′N 103°50′E

For other uses, see Singapore (disambiguation).

Singapore (/ˈsɪŋ(g)əpɔːr/ listen)), officially the **Republic of Singapore**, is a sovereign island city-state in maritime Southeast Asia. It lies about one degree of latitude (137 kilometres or 85 miles) north of the equator, off the southern tip of the Malay Peninsula, bordering the Straits of Malacca to the west, the Riau Islands (Indonesia) to the south, and the South China Sea to the east. The country's territory is composed of one main island, 63 satellite islands and islets, and one outlying islet, the combined area of which has increased by 25% since the country's independence as a result of extensive land reclamation projects. It has the third greatest population density in the world. With a multicultural population and recognising the need to respect cultural identities, Singapore has four official languages; English, Malay, Mandarin, and Tamil. English is the lingua franca. Multiracialism is enshrined in the constitution and continues to shape national policies in education, housing, and politics.

Modern Singapore was founded in 1819 by Sir Stamford Raffles as a trading post of the British Empire. In 1867, the colonies in Southeast Asia were reorganised and Singapore came under the direct control of Britain as part of the Straits Settlements. During the Second World War, Singapore was occupied by Japan in 1942, and returned to British control as a separate crown colony following Japan's surrender in 1945. Singapore gained self-governance in 1959 and in 1963 became part of the new federation of Malaysia, alongside Malaya, North Borneo, and Sarawak. Ideological differences led to Singapore being expelled from the federation two years later and it became an

Republic of Singapore
Three other official names [show]

Flag Coat of arms

Motto: *Majulah Singapura*
(English: "Onward Singapore")

Anthem: *Majulah Singapura*
(English: "Onward Singapore")

0:00 MENU

➜ **Summary:**

"Singapore, officially the Republic of Singapore, is a sovereign island city-state in maritime Southeast Asia."

Unsupervised Content Selection

- Core idea: Finding keywords
  - Choose sentences with many important/informative/salient/etc. word
- Various strategies proposed, e.g.:
  - tf-idf (we already know how to do this)
  - Log-likelihood ratio (LLR)
  - TextRank – graph-based approach
    (supports keyword & sentence extractions)

Log-Likelihood Ratio (LLR)

= important

- Step 1: Identify salient words
  - Assign words with a minimum LLR with a positive weight
  - Option: assign words that are in the query/question with a positive weight

$$weight(w_i) = \begin{cases} 1 & \text{if } -2\log \lambda(w_i) > 10 \\ 1 & \text{if } w_i \in \text{query/question} \\ 0 & \text{otherwise} \end{cases}$$

← In case of query-focused summarization

- Step 2: Score sentences
  - Score if a sentence = average weight over all words in the sentence

$$weight(s) = \frac{1}{|S|} \sum_{w \in S} weight(w)$$

Log-Likelihood Ratio (LLR)

- Underlying assumption
  - Binomial distribution for generating *w* in a text

number of words in text

$$P(\text{word } w \text{ appears } k \text{ times in a text}) = b(p, k, n) = \binom{n}{k} p^k (1-p)^{n-k}$$

counts

probability of w; estimate via MLE: $p = \frac{k}{n}$

counts

- Log-Likelihood Ratio

$$\lambda(w_i) = \frac{b(p, k_d, n_d) \cdot b(p, k_c, n_c)}{b(p_d, k_d, n_d) \cdot b(p_c, k_c, n_c)}$$

probability of observing *w* in document *d* and corpus *c* assuming **equal** probabilities *p* in both *d* and *c*

probability of observing *w* in document *d* and corpus *c* assuming **different** probabilities p_d and p_c in *d* and *c*

TextRank

Core algorithm

1) Identify meaningful text units ➜ set of vertices V
   (either words or sentences depending on task)

2) Identify meaningful relations between text units ➜ set edges E
   (e.g.: co-occurrence of text units ot similarity between text units)

3) Apply graph-based ranking algorithm over G(V, E)
   (proposed in original paper: Weighted PageRank)

4) Sort vertices based on their final score

Represent text as a graph

Important vertex in graph
⇔
Important text unit in document

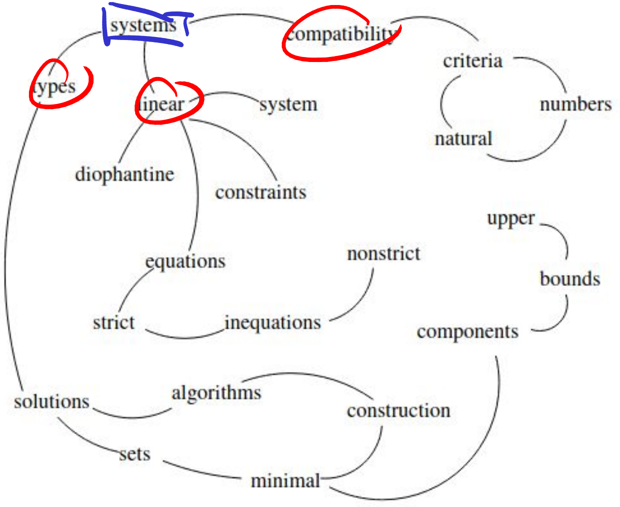
Source: [TextRank: Bringing Order into Texts (2004)]

TextRank

- Identification of keyword
    - Text units = words ➜ vertices = words
    - Unweighted edge = "binary" co-occurrence
      (there exists an edge between to vertices if the two corresponding words appear together within a window)
    - Apply PageRank over resulting Graph
    - Choose keywords with highest scores

Note: PageRank is defined over direct graphs, but an indirect edge can be represented as 2 directed edges.

Compatibility ~~of~~ systems ~~of~~ linear constraints ~~over the~~ set ~~of~~ natural numbers. ~~Criteria of~~ compatibility of a system of linear Diophantine equations, strict inequations, and nonstrict inequations are considered. Upper bounds for components of a minimal set of solutions and algorithms of construction of minimal generating sets of solutions for all types of systems are given. These criteria and the corresponding algorithms for constructing a minimal supporting set of solutions can be used in solving all the considered types systems and systems of mixed types.



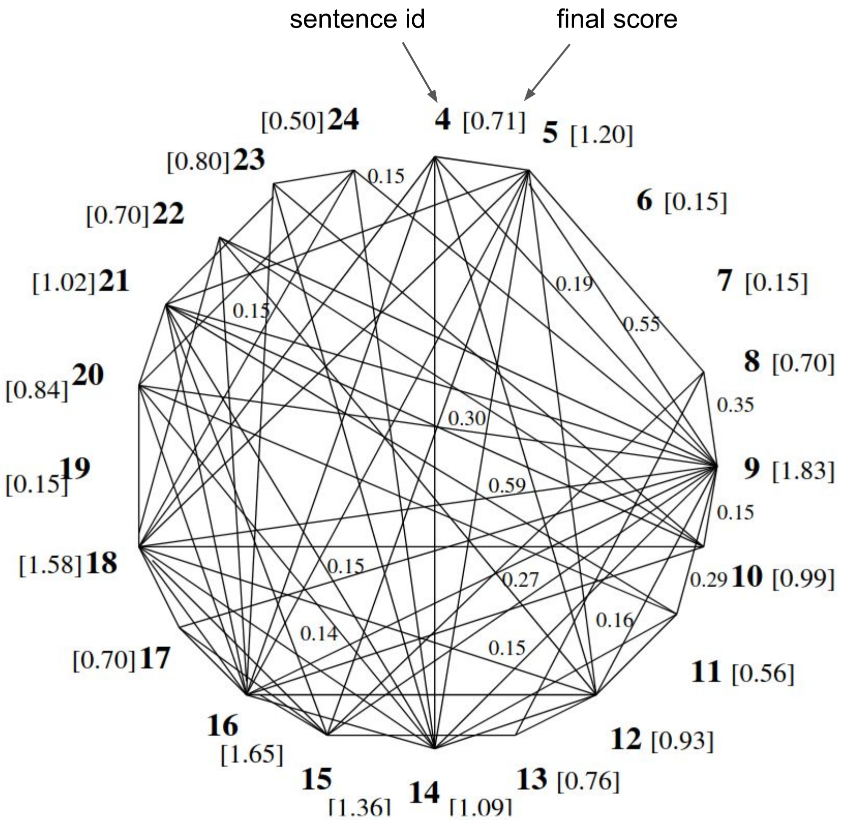
Keywords assigned by TextRank:
linear constraints; linear diophantine equations; natural numbers; nonstrict inequations; strict inequations; upper bounds

Keywords assigned by human annotators:
linear constraints; linear diophantine equations; minimal generating sets; non–strict inequations; set of natural numbers; strict inequations; upper bounds

Source: [TextRank: Bringing Order into Texts (2004)]

TextRank

- Sentence extraction
  - Text units = sentences ➜ vertices = sentences
  - Weighted edge = sentence similarity
    (e.g., Jaccar, cosine between tf-idf / embedding vectors)
  - Apply PageRank over resulting Graph
  - Choose sentences with highest scores

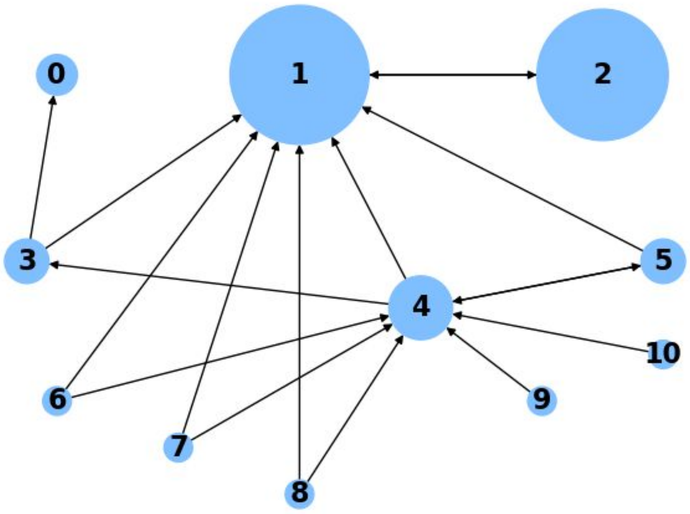


Note: PageRank is defined over unweighted graphs, but can trivially extended to weighted graphs.

Source: [TextRank: Bringing Order into Texts (2004)]

Quick Side Note — PageRank

- PageRank centrality measure
  - Quantifies importance of a node in a graph
    (pages in the Web Graph connected by links)
  - Recursive definition: A node is important if many other important nodes point to it
  - Computing PageRank = Finding the largest Eigenvector of a matrix derived from graph



Quick Quiz

What is the **interpretation** of the text unit (word or sentence) with the **highest** TextRank score?

A This text unit occurs the most frequently in the document

B This text unit has the most connections to all other units

C This text unit appears very early in the document

D This text unit is the best representative for the document

Outline

- **Text Summarization**
  - Overview & Categorization
  - Basic Architecture
  - **Evaluation**
  - Query-Focused Summarization
- Question Answering
  - Overview & Categorization
  - Factoid QA (Basic Architecture)
  - Core Components
  - Extended Concepts

Evaluating Summaries — ROUGE

- ROUGE (Recall Oriented Understudy for Gisting Evaluation)
  - Measure similarity between 2 texts based on n-gram overlap
  - Not as good as human evaluation shown to be a convenient proxy

- Basic procedure: Given a document d and a generated summary $\hat{y}$
  - Have *N* humans produce a set of reference summaries S_H ← reference
  - What percentage of the n-grams from the reference summaries appear in $\hat{y}$?

ROUGE-2

⇒ bigrams

$$\text{ROUGE-N} = \frac{\sum_{S \in S_H} \sum_{g_n \in S} \overbrace{min(Count(g_N, \hat{y}), Count(g_N, S))}^{\text{maximum overlap between reference and candidate w.r.t. n-gram}}}{\sum_{S \in S_H} \sum_{g_N \in S} Count(g_N, S)}$$

specifies of the size of the n-grams to be considered

In-Lecture Activity (5-7 mins)

$$\text{ROUGE-N} = \frac{\sum_{s \in S_H} \sum_{g_N \in \hat{y}} min(Count(g_N, \hat{y}), Count(g_N, S_H))}{\sum_{s \in S_H} \sum_{g_N \in \hat{y}} Count(g_N, S_H)}$$

- Task: Compute the ROUGE-2 score for the example below
  - Post your solution to Canvas > Discussions
    (individually or as a group; include all group members' names in the post)

System-generated summary

$\hat{y}$ *"water spinach is a leaf vegetable commonly eaten in tropical areas of asia"*

3 human-generated summaries (reference)

S_H

"water spinach is a semi-aquatic tropical plant grown as a vegetable" ➜ 10 bigrams

"water spinach is a semi-aquatic tropical plant grown as a vegetable" ➜ 10 bigrams

"water spinach is a commonly eaten leaf vegetable of asia" ➜ 9 bigrams

$Count(g_N, S_H)$

ROUGE — Example (ROUGE-2: bigrams)

System-generated summary

southern

"water spinach is a leaf vegetable commonly eaten in tropical areas of asia"

???

3 human-generated summaries (reference)

"water spinach is a semi-aquatic tropical plant grown as a vegetable" ➜ 10 bigrams

"water spinach is a semi-aquatic tropical plant grown as a vegetable" ➜ 10 bigrams

"water spinach is a commonly eaten leaf vegetable of asia" ➜ 9 bigrams

"water spinach", "spinach is", "is a"

"water spinach", "spinach is", "is a", "commonly eaten", "leaf vegetable', "of asia"

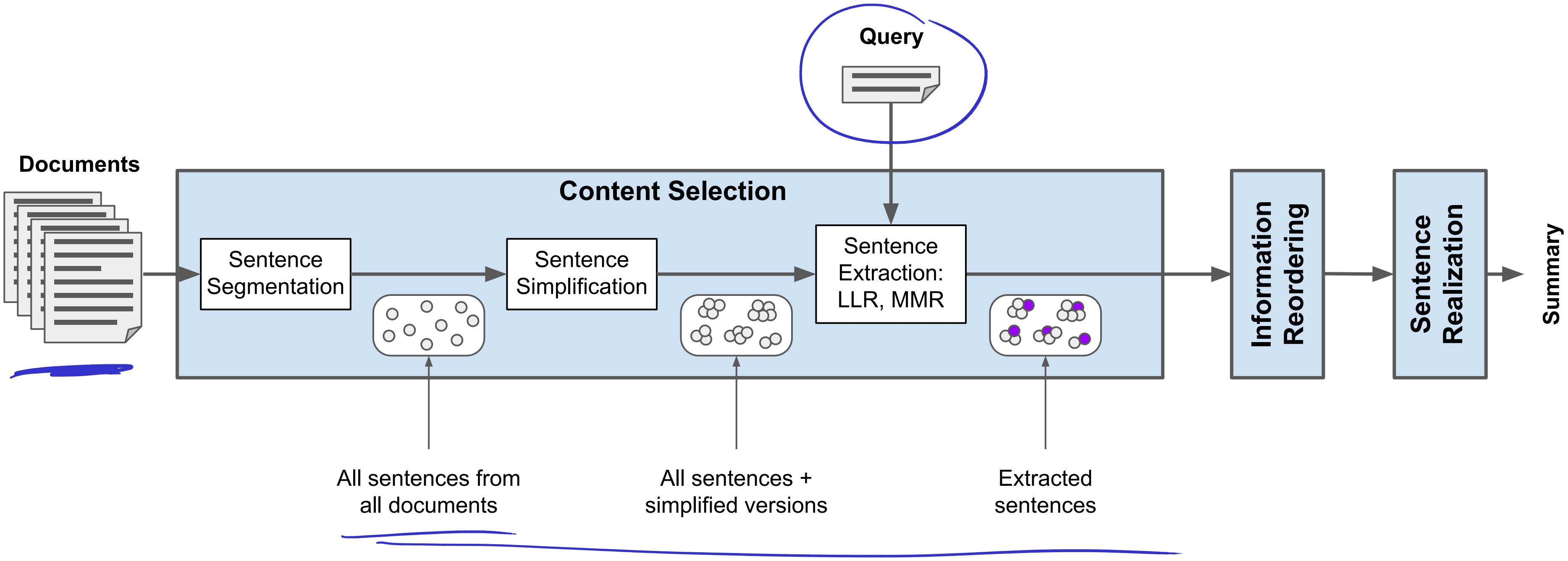
$$\text{ROUGE-2} = \frac{3+3+6}{10+10+9} = 0.4\cancel{3}1$$

Outline

- **Text Summarization**
  - Overview & Categorization
  - Basic Architecture
  - Evaluation
  - **Query-Focused Summarization**

- Question Answering
  - Overview & Categorization
  - Factoid QA (Basic Architecture)
  - Core Components
  - Extended Concepts

Query-Focused Multidocument Summarization



Sentence Simplification

- Unsupervised approach
  - Sentence simplification by sentence trimming
  - Input: parse tree of sentence ➜ trimmed parse tree (dependency parse tree)
    (remove "less important" subtrees based on linguistically-motivated rules)

appositives	*Rajam, ~~28, an artist who was living at the time in Philadelphia,~~ found the inspiration in the back of city magazines.*
attribution clauses	*Rebels agreed to talks with government officials, ~~international observers said Tuesday~~.*
Prepositional phrases without named entities	*The commercial fishing restrictions in Washington will not be lifted unless the salmon population increases ~~to a sustainable number~~.*
initial adverbials	*"For example, [...]", "On the other hand, [...]", "As a matter of fact, [...]", "At this point, [...]"*

MDS Sentence Extraction — Maximal Marginal Relevance (MMR)

- Maximal Marginal Relevance (MMR)
  - Iteratively, greedily pick the best sentence to add to existing summary
    (stop when desired length of summary is reached)
  - 2 selection criteria

    (1) Relevance
    - Sentence s_i is maximally relevant to user's query q
    - Example: high cosine similarity between s_i and q

    (2) Novelty
    - Sentence is minimally redundant to existing summary S so far

Note: Sim1 and Sim2 can be the same similarity measure

$$MMR = \operatorname*{argmax}_{s_i \in \underbrace{C \setminus S}_{\text{all sentence not selected so far}}} \left[\alpha \cdot \underbrace{Sim_1(s_i, q)}_{\text{similarity between } s_i \text{ and query q}} - (1-\alpha) \cdot \underbrace{\max_{s_j \in S} Sim_2(s_i, s_j)}_{\text{max. similarity between } s_i \text{ and an sentence in current summary}} \right]$$

Information Ordering

- Chronological ordering:
  - Order sentences by the date of the document, e.g., for summarizing news
    (Source: Inferring Strategies for Sentence Ordering in Multidocument News Summarization, 2002)

- Coherence:
  - Choose orderings that make neighboring sentences similar (by cosine).
  - Choose orderings in which neighboring sentences discuss the same entity
    (Source: Modeling Local Coherence: An Entity-Based Approach, 2007)

- Topical ordering
  - Learn the ordering of topics in the source documents

Domain-Specific Information Extraction

- Domain: definitions
  - a word's hypernym/genus, synonyms, etc.
- Domain: biographies
  - a person's birth/death, fame factor, education, nationality and so on
- Domain: drugs / drug use
  - **P**roblem (the medical condition)
  - **I**ntervention (the drug or procedure)
  - **C**omparison (e.g., control group)
  - **O**utcome (the result of the study)

PICO

Definitional Templates

- Domain: definitions

hypernym	The Hajj is a type of ritual
synonym	The Hajj, or Pilgrimage of Mecca, is the central duty of Islam
subtype	Qiran, Tamattu's, and Ifrad are three different types of Hajj

- Domain: biographies

dates	was assassinated on April 4, 1968
nationality	was born in Atlanta, Georgia
education	entered Boston University as a doctoral student

- Domain: drugs / drug use

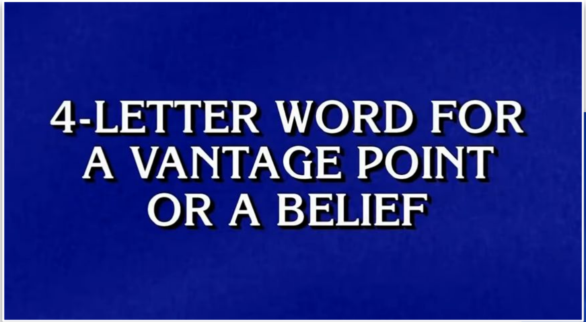
population	37 otherwise healthy children aged 2 to 12 years
intervention	acetaminophen (10 mg/kg)
outcome	ibuprofen provided greater temperature decrement and longer duration of antipyresis than acetaminophen when the two drugs were administered in approximately equal dose

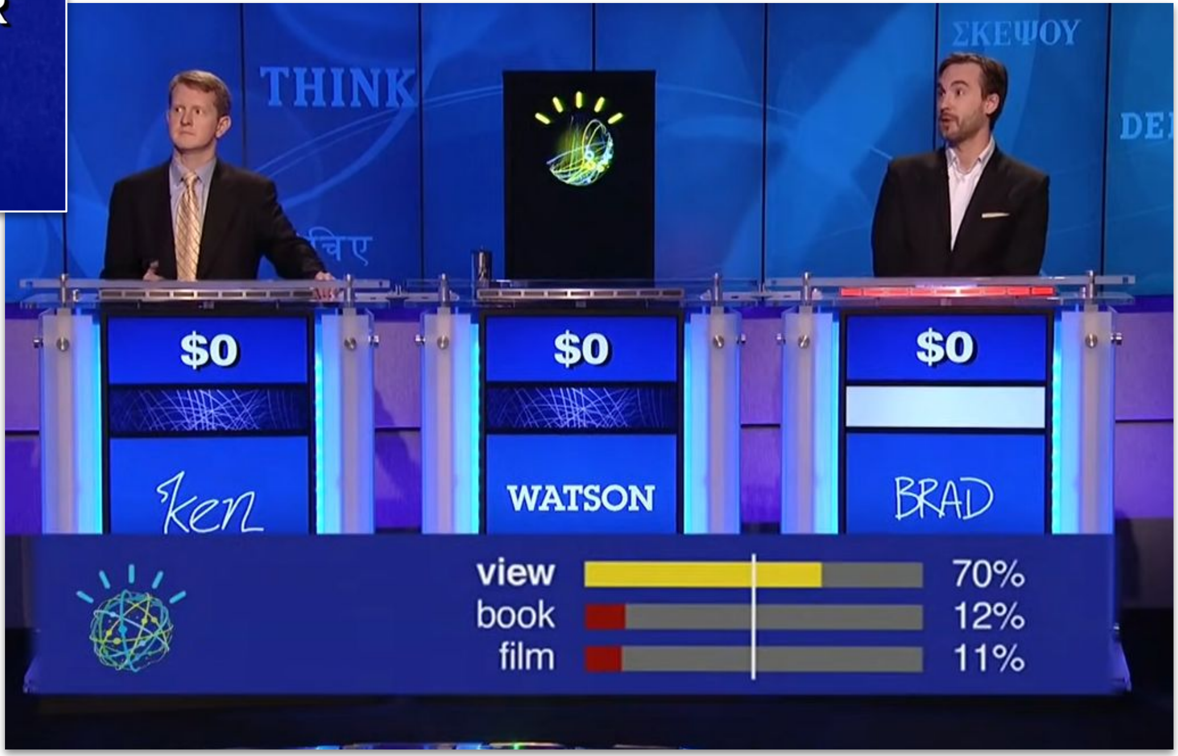
Outline

- Text Summarization
  - Overview & Categorization
  - Basic Architecture
  - Evaluation
  - Query-Focused Summarization

- **Question Answering**
  - **Overview & Categorization**
  - Factoid QA (Basic Architecture)
  - Core Components
  - Extended Concepts

Rise of the Machines

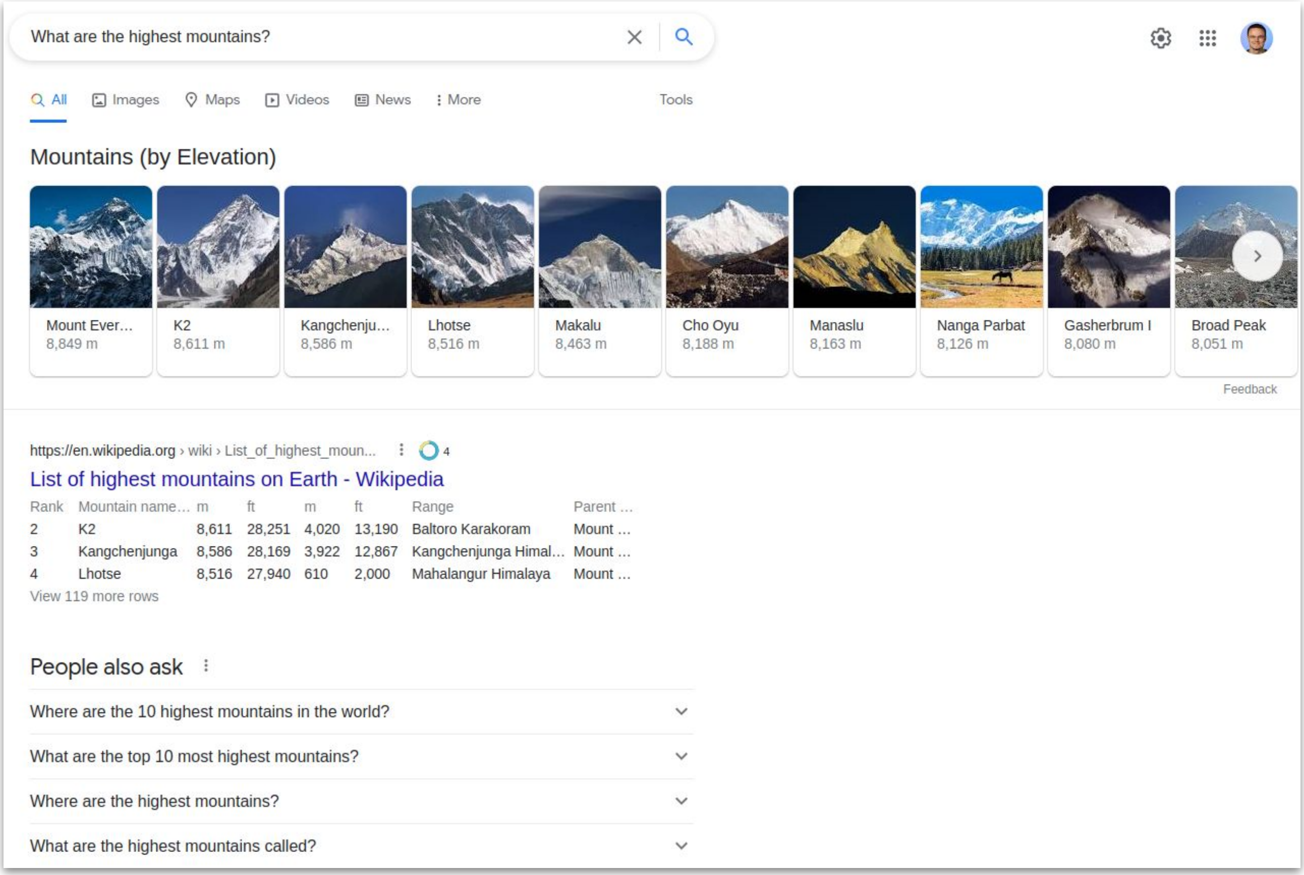




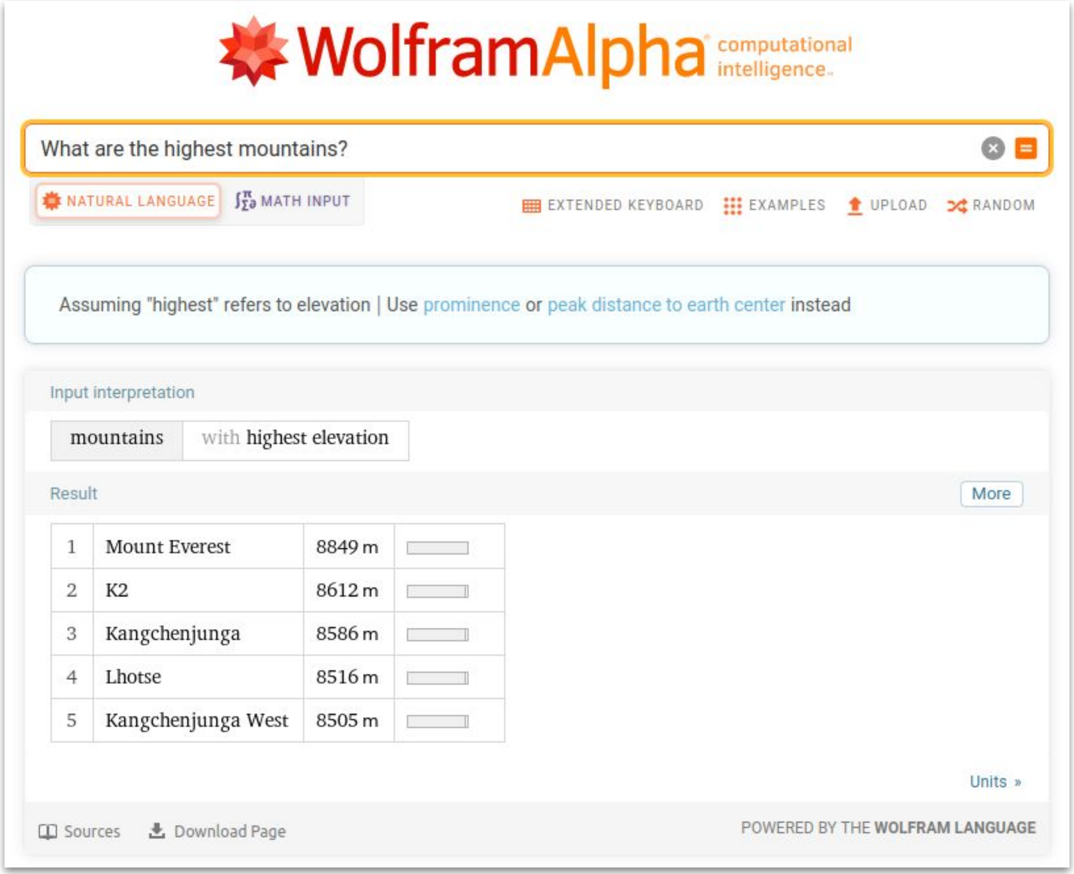
IBM Watson won Jeopardy!
on February 16, 2011

Source: [Watson and the Jeopardy! Challenge]

Question Answering via Web Search



Question Answering via Web Search



The Latest King in Town: GPT-x / ChatGPT

You are playing Jeopardy, and the answer is "4-letter word for a vantage point or a belief". What is the correct questions?

What is the word "View"?

Question Answering — Dimensions

Context / Source

- Passage, document, corpus, …, the Web
- Knowledge base / knowledge graphs
- Semi-structured tables
- Images / Video
- ...combination of sources

Question Types

- Factoid questions
  (typically direct and clear answers)

  "How many calories does a tub of Ben & Jerry's have?"

- Open-ended questions
  (narratives, opinions, descriptions, etc.)

  "What is the healthiest way to quickly lose weight?"

Answer Types

- Yes/No
- Short text span/paragraph
  (extracted or generated)
- Database entry
- List of alternatives
- generation

Question Answering — Datasets

- SQuAD (2016)
  - Stanford Question Answering Dataset
  - Over 100k and questions & answers generated by crowdworkers

Source: [SQuAD: 100,000+ Questions for Machine Comprehension of Text]

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under gravity. The main forms of precipitation include drizzle, rain, sleet, snow, graupel and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals within a cloud. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

- SQuAD 2.0 (2018)
  - Over 50k+ question & answers (crowdsourced)
  - Twist: unanswerable question & plausible answers

Source: [Know What You Don't Know: Unanswerable Questions for SQuAD]

Article: Endangered Species Act
Paragraph: "*. . . Other legislation followed, including the Migratory Bird Conservation Act of 1929, a 1937 treaty prohibiting the hunting of right and gray whales, and the Bald Eagle Protection Act of 1940. These later laws had a low cost to society—the species were relatively rare—and little opposition was raised.*"

Question 1: "*Which laws faced significant opposition?*"
Plausible Answer: *later laws*

Question 2: "*What was the name of the 1937 treaty?*"
Plausible Answer: *Bald Eagle Protection Act*

Question Answering — Datasets

- MCTest (2013)
  - MCT: machine comprehension of text
  - Generation of dataset done by crowdworkers
    (short stories + factoid questions with 4 multiple choice answers + opened-ended (more chall

Source: [MCTest: A Challenge Dataset for the Open-Domain](#) [Machine Comprehension of Text](#)

Context/
source

James the Turtle was always getting in trouble. Sometimes he'd reach into the freezer and empty out all the food. Other times he'd sled on the deck and get a splinter. His aunt Jane tried as hard as she could to keep him out of trouble, but he was sneaky and got into lots of trouble behind her back.

One day, James thought he would go into town and see what kind of trouble he could get into. He went to the grocery store and pulled all the pudding off the shelves and ate two jars. Then he walked to the fast food restaurant and ordered 15 bags of fries. He didn't pay, and instead headed home.

His aunt was waiting for him in his room. She told James that she loved him, but he would have to start acting like a well-behaved turtle.

After about a month, and after getting into lots of trouble, James finally made up his mind to be a better turtle.

1) What is the name of the trouble making turtle?
A) Fries
B) Pudding
C) James
D) Jane

2) What did James pull off of the shelves in the grocery store?
A) pudding
B) fries
C) food
D) splinters

3) Where did James go after he went to the grocery store?
A) his deck
B) his freezer
C) a fast food restaurant
D) his room

Question Answering — Datasets

- CoQA (2019)
  - CoQA: Conversational Question Answering
  - Dataset generation by pairs of crowdworkers
    (one asking the questions, one answering the questions)
  - 127k questions with answers from 8k conversations

Source: [CoQA: A Conversational Question Answering Challenge]()

The Virginia governor's race, billed as the marquee battle of an otherwise anticlimactic 2013 election cycle, is shaping up to be a foregone conclusion. Democrat Terry McAuliffe, the longtime political fixer and moneyman, hasn't trailed in a poll since May. Barring a political miracle, Republican Ken Cuccinelli will be delivering a concession speech on Tuesday evening in Richmond. In recent ...

Q_1: What are the candidates **running** for?
A_1: Governor
R_1: The Virginia governor's race

Q_2: **Where**?
A_2: Virginia
R_2: The Virginia governor's race

Q_3: Who is the democratic candidate?
A_3: **Terry McAuliffe**
R_3: Democrat Terry McAuliffe

Q_4: Who is **his** opponent?
A_4: **Ken Cuccinelli**
R_4 Republican Ken Cuccinelli

Q_5: What party does **he** belong to?
A_5: Republican
R_5: Republican Ken Cuccinelli

Q_6: Which of **them** is winning?
A_6: Terry McAuliffe
R_6: Democrat Terry McAuliffe, the longtime political fixer and moneyman, hasn't trailed in a poll since May

Question Answering — Datasets

Dataset	Conversational	Answer Type	Domain
MCTest (Richardson et al., 2013)	✗	Multiple choice	Children's stories
CNN/Daily Mail (Hermann et al., 2015)	✗	Spans	News
Children's book test (Hill et al., 2016)	✗	Multiple choice	Children's stories
SQuAD (Rajpurkar et al., 2016)	✗	Spans	Wikipedia
MS MARCO (Nguyen et al., 2016)	✗	Free-form text, Unanswerable	Web Search
NewsQA (Trischler et al., 2017)	✗	Spans	News
SearchQA (Dunn et al., 2017)	✗	Spans	Jeopardy
TriviaQA (Joshi et al., 2017)	✗	Spans	Trivia
RACE (Lai et al., 2017)	✗	Multiple choice	Mid/High School Exams
Narrative QA (Kočiskỳ et al., 2018)	✗	Free-form text	Movie Scripts, Literature
SQuAD 2.0 (Rajpurkar et al., 2018)	✗	Spans, Unanswerable	Wikipedia

Source: CoQA: A Conversational Question Answering Challenge (2019)

Question Answering — Datasets

- Long Form Question Answering
  - Question & answers collected from Subreddit ELI5
    (Explain Like I'm 5: https://www.reddit.com/r/explainlikeimfive/)
  - Collecting of source document from Common Crawl
    (https://commoncrawl.org/)
  - 272k questions with at least 1 answer each

Source: ELI5: Long Form Question Answering (2019)

Question: How do Jellyfish function without brains or nervous systems? [...] (60 words)

Answer: Jellyfish may not have a brain, but they have a rough nervous system and innate behaviours. However, they are very simple creatures. They're invertebrate: creatures without a backbone. Most jellyfish have really short life spans. Sometimes just a couple of hours. [...] As their name implies, they are largely composed of basically jelly inside a thin membrane. They're over 95% water. (327 words)

Documents: [...] Jellyfish do not have brains, and most barely have nervous systems. They have primitive nerve cells that help them orient themselves in the water and sense light and touch. [...] While they dont possess brains, the animals still have neurons that send all sorts of signals throughout their body. [...] They may accomplish this through the assistance of their nerve rings. Jellyfish don't have brains, and that's just where things begin. They don't have many of the body parts that are typical in other animals. [...] (1070 words)

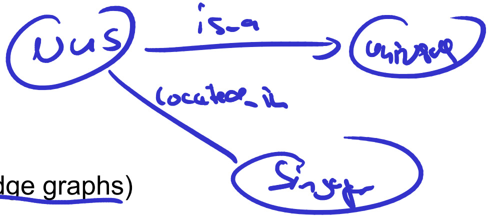
Outline

- Text Summarization
  - Overview & Categorization
  - Basic Architecture
  - Evaluation
  - Query-Focused Summarization

- **Question Answering**
  - Overview & Categorization
  - **Factoid QA (Basic Architecture)**
  - Core Components
  - Extended Concepts

QA Systems — Main Paradigms

- Information retrieval-based QA systems
  - Built on top of large text corpora (unstructured data)
  - Use IR techniques find relevant passages (or documents)
  - Apply reading comprehension algorithms over passages and draw answer (algorithms can be feature-based, neural-based, or both)



- Knowledge-based QA systems
  - Built on top of semantic representations (structured data, e.g., knowledge graphs)
  - Parse question to predicate calculus (e.g., FOL) or a query language (e.g., SQL, SPARQL)
  - Optional: Generate "nice" answer from results

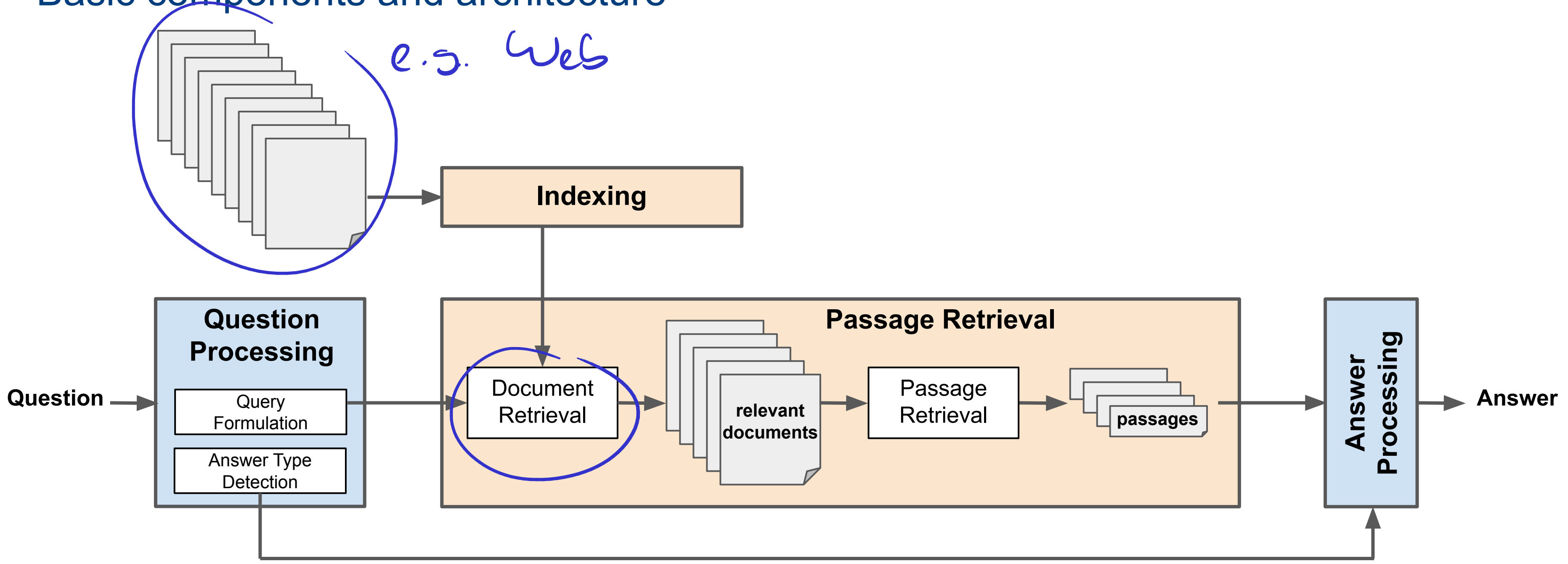
- Hybrid Q&A systems

IR-Based Factoid QA Systems

(easier)

Basic components and architecture

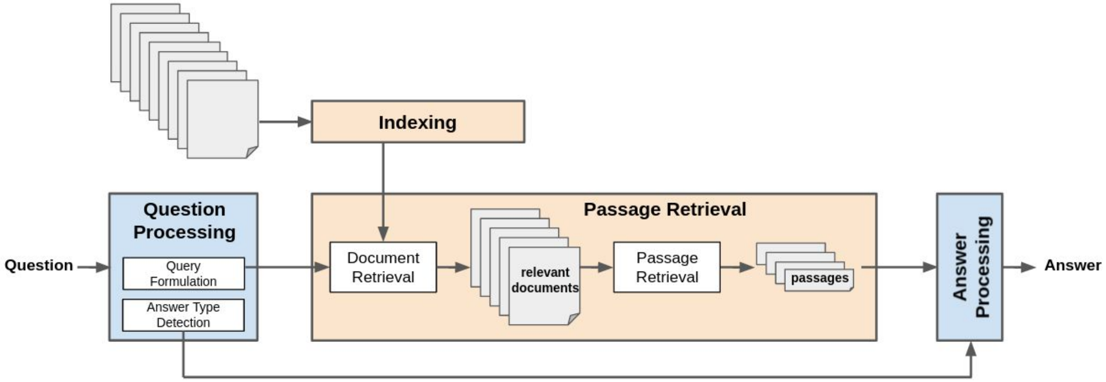
e.g. Web



Source: [Speech and Language Processing, Chapter 25]

IR-Based Factoid QA Systems

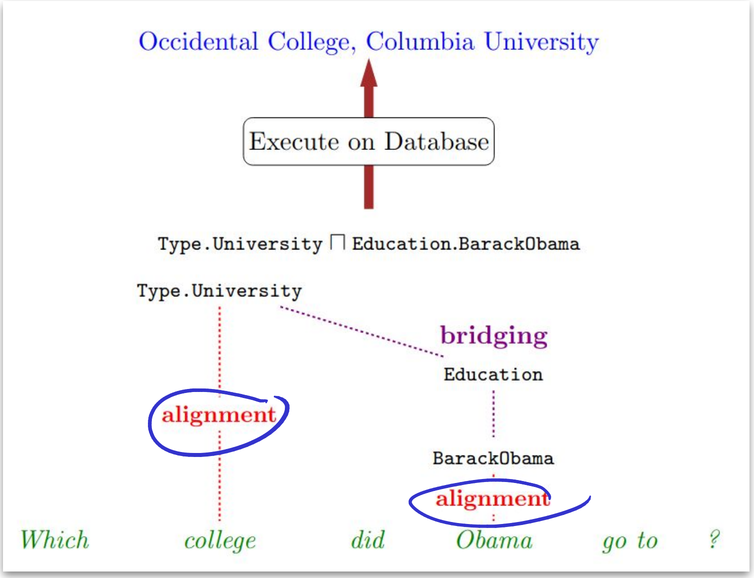
- Question Processing
  - Detect question type, answer type, focus, relations
  - Formulate queries to send to a search engine / database
- Passage Retrieval
  - Retrieve ranked documents
  - Break into suitable passages and rerank
- Answer Processing
  - Extract candidate answers
  - Rank candidates using evidence from the text and external sources



Knowledge-Based Factoid QA Systems

- Build semantic representation of question
  - times, dates, locations, entities, numeric quantities, etc.

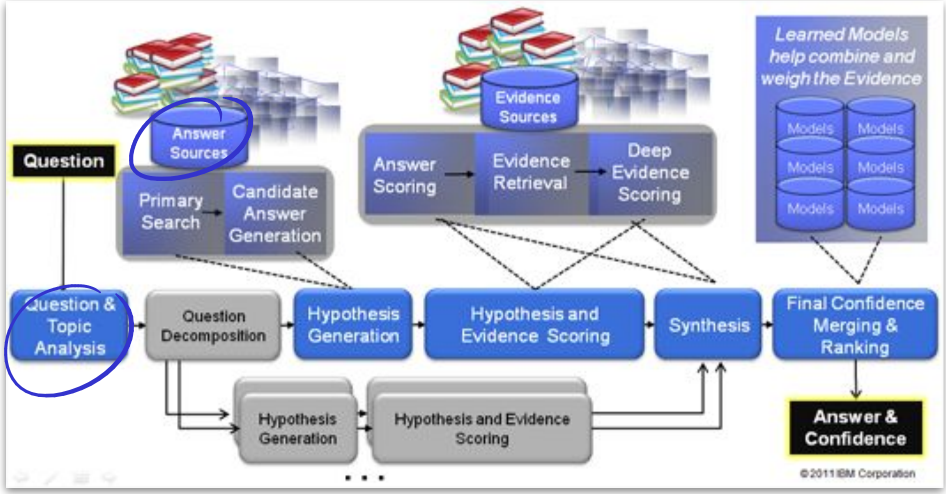
- Use representations to query structured data or resources
  - Geospatial databases
  - Ontologies (Wikipedia Infoboxes, dbPedia, WordNet, Yago)
  - Scientific databases
  - etc.



Source: [Semantic Parsing on Freebase from Question-Answer Pairs (2013)](#) (Freebase knowledge graph no longer available)

Hybrid QA Systems

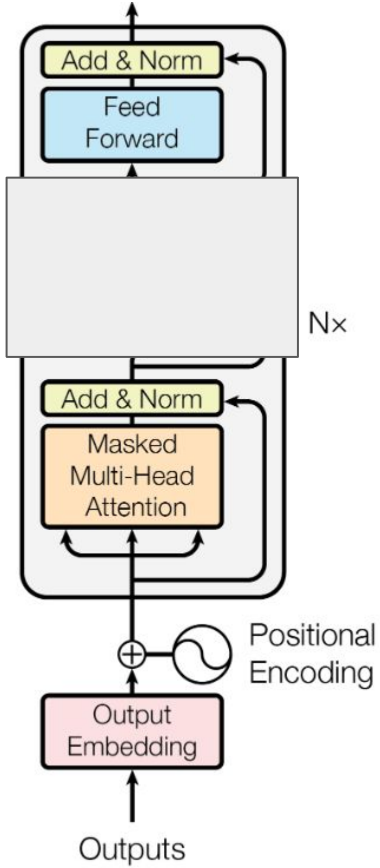
- Example: IBM Watson
  - Build a shallow semantic representation of the query
  - Generate answer candidates using IR methods
    (Augmented with ontologies and semi-structured data)
  - Score each candidate using richer knowledge sources
    (geospatial databases, temporal reasoning, taxonomical classification)



Source: [IBM Website]

QA using Large Language Models: GPT (Generative Pretrained Transformer)

- GPT
  - Uses only the Transformer Decoder without the encoder attention block (alternatively: encoder with "do not look ahead" masking)
  - Self-supervised training
- Learning objectives
  - Auto-regressive Language Model
- (Very) oversimplified history of GPT
  - GPT-1/2/3: text only, "just" making it larger; GPT-4: multimodal
  - GPT-3+: **reinforcement learning from human feedback** (RLHF)



Outline

- Fake News Detection
  - Guest lecture by Chieu Hai Leong (DSO National Laboratories)
- Text Summarization
  - Overview & Categorization
  - Basic Architecture
  - Evaluation
  - Query-Focused Summarization
- **Question Answering**
  - Overview & Categorization
  - Factoid QA (Basic Architecture)
  - **Core Components**
  - Extended Concepts

Question Processing

- Things to extract from the question:
  - Answer Type Detection (decide the named entity type (e.g., person, place) of the answer)
  - Query Formulation (choose query keywords for the IR system)
  - Question Type classification (factoid question? definition question? math question? etc?)
  - Focus Detection (find the question words that are replaced by the answer)
  - Relation Extraction (find relations between entities in the question)

HUMAN. individual

"Who was the first president of Singapore?"

"who" ➜ factoid question

Question word: ***"who"***
Answer is a person (name)

important keywords

Answer Type Taxonomy (Li & Roth, 2002)

- 2- layered taxonomy
  - 6 coarse classes
    (ABBREVIATION, ENTITY, DESCRIPTION, HUMAN, LOCATION and NUMERIC VALUE)
  - 50 fine classes
  - On the right: distribution of 500 questions in TREC-10 Question Classification test dataset

who

where

Class	#	Class	#
ABBREV.	9	description	7
abb	1	manner	2
exp	8	reason	6
ENTITY	94	**HUMAN**	65
animal	16	group	6
body	2	individual	55
color	10	title	1
creative	0	description	3
currency	6	**LOCATION**	81
dis.med.	2	city	18
event	2	country	3
food	4	mountain	3
instrument	1	other	50
lang	2	state	7
letter	0	**NUMERIC**	113
other	12	code	0
plant	5	count	9
product	4	date	47
religion	0	distance	16
sport	1	money	3
substance	15	order	0
symbol	0	other	12
technique	1	period	8
term	7	percent	3
vehicle	4	speed	6
word	0	temp	5
DESCRIPTION	138	size	0
definition	123	weight	4

Source: Learning Question Classifiers (2002)

Answer Type Detection

- Hand-written rules, e.g.:
  - Regular Expressions
  - Question headword
    (first noun phrase after the wh-word)
    who/what/where/when/~~why~~
    which
- Machine Learning
  - Requires annotated question datasets
  - Train classifier(s) over annotated dataset
    (feature-based, neural-based, or both)
  - Wide range of relevant features
    (question words, POS tags, parse features, named entities, etc.)
- Hybrid Methods

"Which **city** in Asia is also called the Garden City?"

"What is the official **mascot** of Singapore."

Question Processing

Query Formulation

Answer Type Detection

Query Formulation — Keyword Selection

Question Processing

Query Formulation

Answer Type Detection

- Keyword heuristics (ordered list!)
  - (1) Select all non-stop words in quotations

    proper nouns
  - (2) Select all NNP words in recognized named entities
  - (3) Select all complex nominals with their adjectival modifiers
  - (4) Select all other complex nominals
  - (5) Select all nouns with their adjectival modifiers
  - (6) Select all other nouns
  - (7) Select all verbs
  - (8) Select all adverbs
  - (9) Select the question focus word(s)

    (skipped in all previous steps)
  - (10) Select all other words

Source: The structure and Performance of an Open-Domain Question Answering System (2000)

Query Formulation — Keyword Selection

Question Processing

Query Formulation

Answer Type Detection

(4) Select all other complex nominals

(1) Select all non-stop words in quotations

(2) Select all NNP words in recognized named entities

"Who played the lead in the movie "Dune" directed by Denis Villeneuve?"

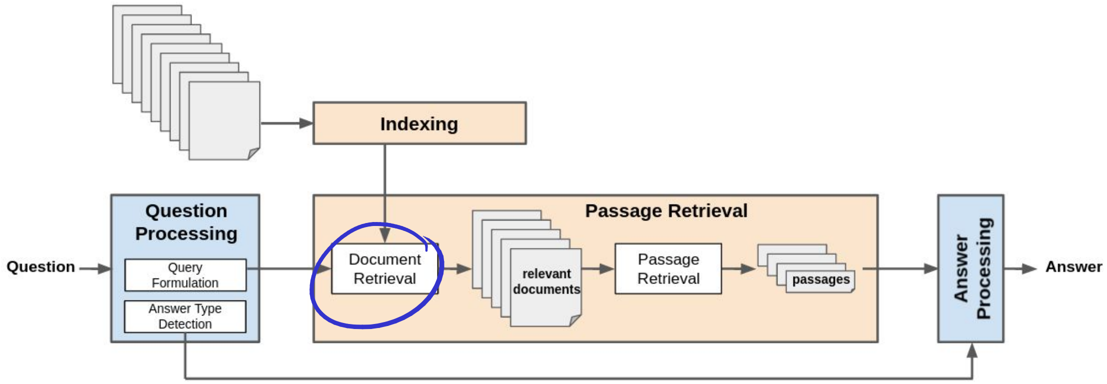
(7) Select all verbs

➜ **Output:** Dune/1, Denis Villeneuve/2, lead/4, movie/4, played/7, directed/7

Outline

- Fake News Detection
  - Guest lecture by Chieu Hai Leong (DSO National Laboratories)
- Text Summarization
  - Overview & Categorization
  - Basic Architecture
  - Evaluation
  - Query-Focused Summarization
- **Question Answering**
  - Overview & Categorization
  - Factoid QA (Basic Architecture)
  - **Core Components**
  - Extended Concepts

Passage Retrieval



- IR engine retrieves documents using query terms
- Segment the documents into shorter units
  - Typically paragraphs, sentences, text spans
- Passage ranking
  - Use answer type to help re-rank passages

Passage Ranking

- Common features
  - Number of Named Entities of the right type in passage
  - Number of query words in passage
  - Number of question N-grams also in passage
  - Proximity of query keywords to each other in passage
  - Longest sequence of question words
  - Rank of the document containing passage

Answer Extractions

- Answer extraction — core task
  - Extract a specific answer from the passage (typically multiple answer candidates)
  - **Span labeling**: given a passage, identifying span of text which constitutes an answer
- Different strategies
  - Simple baseline: Run NER tagger on passage and return span in the passage is with correct answer type
  - Feature-based answer extraction
  - Neural-based answer extraction

detected answer type

"Who was the first president of Singapore?" (**PERSON**)

"**Yusof bin Ishak** was a Singaporean politician who was the first president of **Singapore**, serving from 1965 to 1970."

NER tag: PERSON

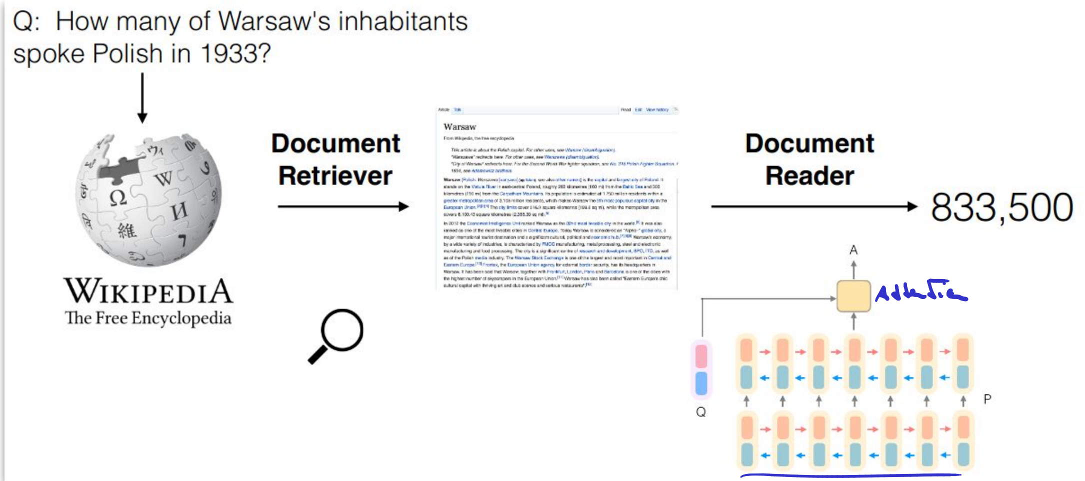
NER tag: GPE

Feature-Based Answer Extraction

- Common features
  - Answer type match (candidate contains a phrase with the correct answer type) → NER type
  - Pattern match (regular expression pattern matches the candidate)
  - Question keywords (number of question keywords in the candidate)
  - Keyword distance (distance in words between the candidate and query keywords)
  - Novelty factor (a word in the candidate is not in the query)
  - Apposition features (candidate is an appositive to question terms)
  - Punctuation location (candidate is immediately followed by a comma, period, quotation marks, semicolon, or exclamation mark)
  - Sequences of question terms (the length of the longest sequence of question terms that occurs in the candidate answer)

Neural-Based Answer Extraction

- Example: DrQA



Document Retriever

- Basic IR-based approach
- Articles and questions are compared as TF-IDF weighted BoW vectors

Document Reader

- Vector representations of questions and paragraphs using RNN encoder
- Train 2 independent classifiers over encoded question and paragraphs

  (1) Predict start of answer span

  (2) Predict end of answer span

Source: [Reading Wikipedia to Answer Open-Domain Questions (2017)]

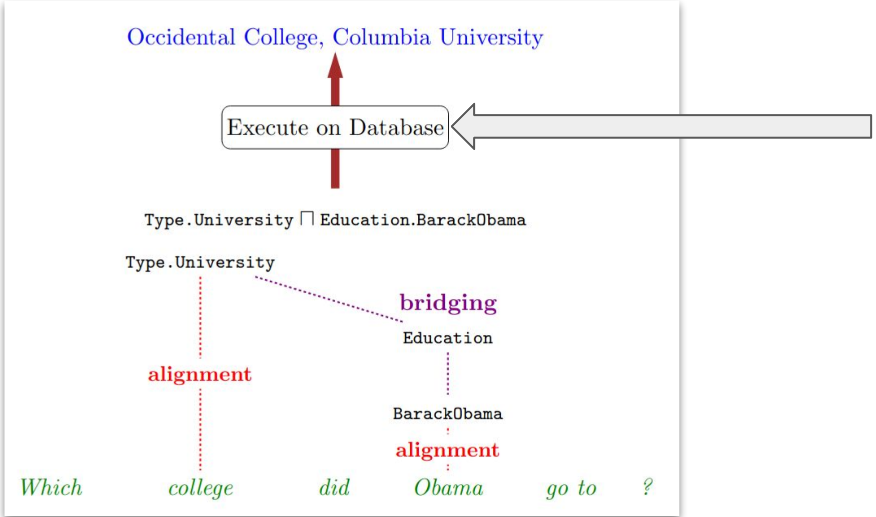
Outline

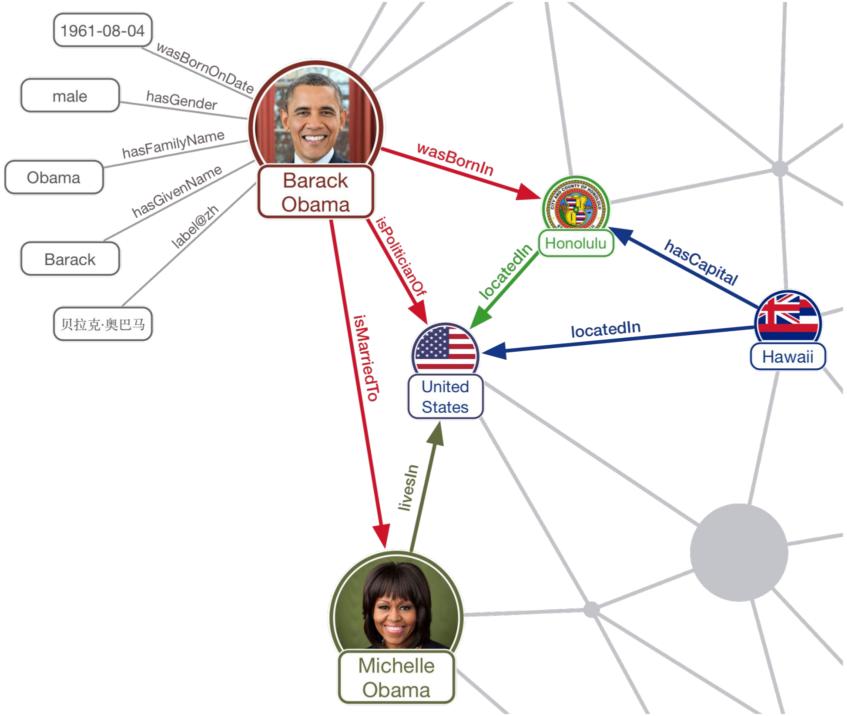
- Text Summarization
  - Overview & Categorization
  - Basic Architecture
  - Evaluation
  - Query-Focused Summarization

- **Question Answering**
  - Overview & Categorization
  - Factoid QA (Basic Architecture)
  - Core Components
  - **Extended Concepts**

Knowledge-Based QA Systems

- Information source: knowledge graphs
  - Structured representation of knowledge
  - e.g.: DBpedia, Wikidata, YAGO, NELL, etc.





Source: [Semantic Parsing on Freebase from Question-Answer Pairs (2013)] (Freebase knowledge graph no longer available)

Knowledge-Based Factoid QA Systems

- Knowledge graph: database of relations
  - (Semi-)automatic extractions from public data sources
    (often manually curated sources, e.g., Wikipedia infoboxes)
  - Relation extraction from unstructured text corpora
    (tricky task, many research papers)

$$\begin{aligned}
&\text{Spouse}(\text{Barack_Obama}, \text{Michelle_Robinson})\\
&\text{Occupation}(\text{Barack_Obama}, \text{Politician})\\
&\text{Occupation}(\text{Barack_Obama}, \text{Lawyer})\\
&\text{GraduatedFrom}(\text{Barack_Obama}, \text{Columbia_University})
\end{aligned}$$

- Extraction relations in questions
  - e.g., meaning representation with FOL
    (question ➜ FOL expression typically contains variables)

"What college did Obama go to?" ➜ $\text{GraduatedFrom}(\text{Barack\_Obama}, x)$

Geospatial Knowledge

- Knowledge about containment, overlap, directionality, borders, e.g.:
  - "Singapore" a possible answer for "Asian city"
  - "Woodlands" is an area/zone/region in "Singapore"

GeoNames knowledge graph

	Name	Country	Feature class	Latitude	Longitude
1	Singapore SIN,Sin-ka-po,Singapore,Singapore City,Singapour,Singapur,Singapura,Sinkapoure,Sìn-kâ-po,Tumasik,cin...	Singapore, SG.01	capital of a political entity population 3,547,809	N 1° 17' 22''	E 103° 51' 0''
2	Singapore Cingapura,Republic of Singapore,Sigapoa,Singaboor,Singaepuru,Singapo,Singapoa,Singapoer,Singapoo,Sin...	Singapore,	independent political entity population 5,638,676	N 1° 22' 0''	E 103° 48' 0''
3	Singapore Changi Airport Aerodrom Singapur,Aeroport Changi,Aeroport Internacional de Singapur-Changi,Aeroport de Singapour Ch...	Singapore, SG.02	airport elevation 6m	N 1° 21' 18''	E 103° 59' 24''
4	Central Singapore Community Development Council Central Singapore,Centre,jungbu sing-gapoleu jigu sahoe baljeon isahoe,shingaporu zhong yang she hui...	Singapore, SG.01	region	N 1° 17' 55''	E 103° 51' 13''
5	Woodlands Woodlands,Woodlands New Town	Singapore, SG.03	populated place population 252,530	N 1° 26' 16''	E 103° 47' 19''
6	National University of Singapore GJai hoc Quoc gia Singapore,Nacional'nij universitet Singapuru,Nacional'nyj universitet Singapura,Na...	Singapore,	college	N 1° 17' 46''	E 103° 46' 47''
7	Singapore River Rio Singapur,Riviere Singapour,Rivière Singapour,Río Singapur,Sin-ka-pho Ho,Sin-ka-pho Hô,Singapore,...	Singapore,	stream	N 1° 17' 12''	E 103° 51' 9''
8	Universal Studios Singapore	Singapore	amusement park	N 1° 15' 20''	E 103° 49' 15''
9	Singapore Botanic Gardens Botanic Gardens,Kebun Botani Singapura,Singapore Botanic Gardens,Taman Botanik Singapura,xin jia po ...	Singapore,	nature reserve	N 1° 18' 37''	E 103° 48' 59''

Temporal Reasoning

- Common observation
  - Answers depend on current time or time frame
  - Common attribute in many knowledge graphs
    (also interesting: biographical dictionaries, obituaries, etc.)

- Example from IBM Watson

“In 1594 he took a job as a tax collector in Andalusia”

Candidate answers
- "Thoreau" is a bad answer (born in 1817)
- "Cervantes" is possible (was alive in 1594)

Context and Conversation in Virtual Assistants

- Coreference helps resolve ambiguities
  - Question focus outside the actual question

Alice: *"Book a flight to* ***Singapore*** *for next Tuesday!"*

Alice: *"What's* ***its*** *timezone?"*

- Clarification questions
  - Insufficient information to find answer
  - Too many possible answer candidates

Alice:	*"Does Fullerton have rooms available on the weekend?"*
Siri:	*"Do you mean the Fullerton Hotel or the Fullerton Bay Hotel?"*

Common Evaluation Metrics

- Accuracy
  - Does answer match gold-labeled answer?
  - Often to "harsh" since an answer might be partially correct
- Mean Reciprocal Rank (MRR)
  - For each questions, return a ranked list of M candidate answers.
  - Question score is 1/*Rank* of the first correct answer

    if 1st answer is correct: 1.0 ← best/ideal case
    else if 2nd answer is correct: 1/2
    else if 3rd answer is correct: 1/3
    …
    else if M-th answer is correct: 1/M
    else: 0 (none of the M answers is correct)

  - Take mean over scores for all N questions

Summary

- Classification as core task of "higher-level" NLP applications
  - Often in combination with different core tasks (e.g., information retrieval, document ranking, etc.)

(1) **Fake News Detection (Assignment 2)**
- Predict if a document (e.g., news article, tweet) is fake

(2) **Text Summarization**
- Predict if a sentence is relevant to be part of a summary

(3) **Question Answering**
- Predict question and answer type
- Feature-based answer extraction

Pre-Lecture Activity

- Assigned Task
  - Do a web search and for the question stated below
  - Post you answer(s) to the question into the Discussion on Canvas (please cite or quote your sources)

“What are current limitations and challenges of LLMs (and using LLMs)?”

Side notes:

- This task is meant as a warm-up to provide some context for the next lecture
- No worries if you get lost; we will talk about this in the next lecture
- You can just copy-&-paste others' answers but his won't help you learn better

Solutions to Quick Quizzes

- Slide 22: D
    - Recall the edge weights reflect the similarity between sentences
    - Sentence with a high Text Rank = sentence similar to many other sentences