

Faculty for Informatics

Technical
University
of Munich



Natural Language Processing

IN2361

PD Dr. Georg Groh

Social Computing
Research Group

Chapter 25

Question Answering

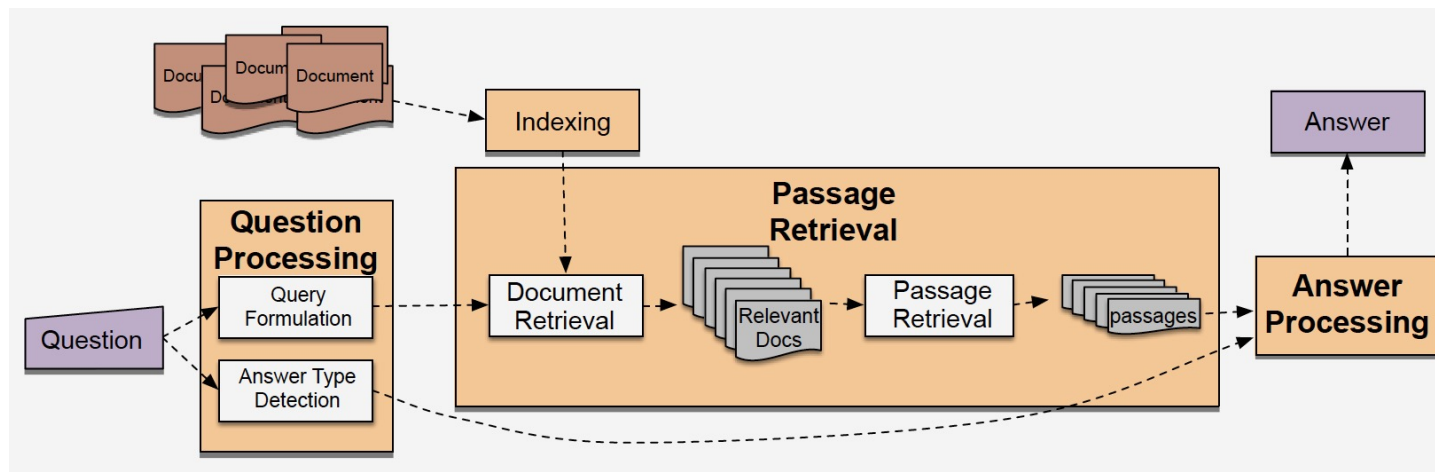
- content is based on [1]
- certain elements (e.g. equations or tables) were taken over or taken over in a modified form from [1]
- citations of [1] or from [1] are omitted for legibility
- errors are fully in the responsibility of Georg Groh
- BIG thanks to Dan and James for a great book!

IR-based Factoid Question Answering

- **factoid** question answering via **Information Retrieval** (i.e. by finding as answers short text segments on the Web or some other collection of documents)

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
How many pounds are there in a stone?	14

- **three phases**: question processing, passage retrieval and ranking, and answer processing



goal: **extract information** from question:

- **answer type**: person, location, time, etc.
- **query**: keywords for the IR system
- **focus**: string of words in the question that are likely to be replaced by the answer
- **question type**: definition question, math question, list question, etc.

example: *Which US state capital has the largest population?*

- answer type: city
- query: US state capital, largest, population
- focus: state capital

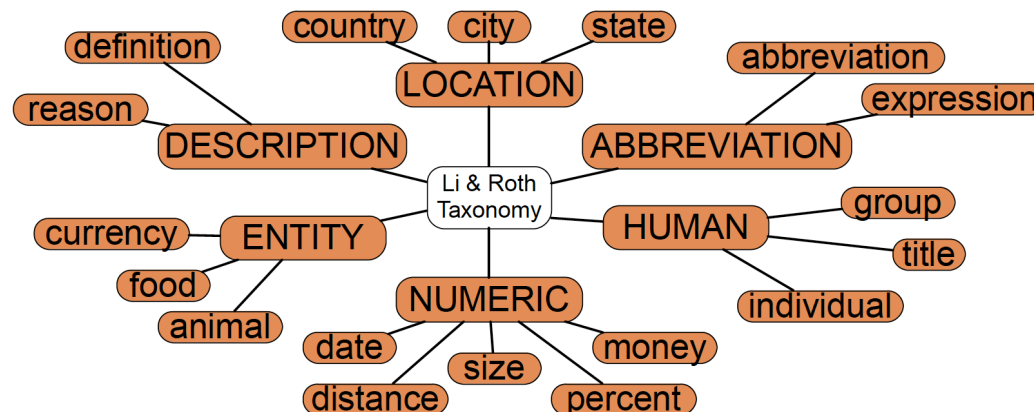
question → **set of** (token-) **keywords** as **query** for IR system

- basic alternatives:
 - basically use all words in question
 - with or without stop-word removal
 - use only NPs
 - with or without question-words (*where, who, when,...*)
- use **query expansion** (especially on smaller document collections where only few specialized answer variants may exist (in contrast to the Web)):
 - expand with all morphological variants of the words,
 - expand with synonyms etc.

- **reformulate query** as part of answer to match respective parts of documents in document collection:
 - **examples:**
 - when was the laser invented? → the laser was invented*
 - where is the Valley of the Kings? → the Valley of the Kings is located in*
 - use **rules** such as:
 - wh-word *did* A verb B → A verb+ed B
 - where is* A → A is located in

Answer Type Detection (Question Classification)

- classify: input: question; output: **type** of the expected **answer**:
Who founded Virgin Airlines? → PERSON.
What Canadian city has the largest population? → CITY.
- **type system**: named entity types or from larger hierarchy (**answer type taxonomy**)



- classifier: **hand built rules** or **machine learning**.

Answer Type Detection (Question Classification)

answer type taxonomy by Li and Roth (2002)

Tag	Example		
ABBREVIATION		HUMAN	
abb	What's the abbreviation for limited partnership?	description	Who was Confucius?
exp	What does the "c" stand for in the equation $E=mc^2$?	group	What are the major companies that are part of Dow Jones?
DESCRIPTION		ind	Who was the first Russian astronaut to do a spacewalk?
definition	What are tannins?	title	What was Queen Victoria's title regarding India?
description	What are the words to the Canadian National anthem?	LOCATION	
manner	How can you get rust stains out of clothing?	city	What's the oldest capital city in the Americas?
reason	What caused the Titanic to sink ?	country	What country borders the most others?
ENTITY		mountain	What is the highest peak in Africa?
animal	What are the names of Odin's ravens?	other	What river runs through Liverpool?
body	What part of your body contains the corpus callosum?	state	What states do not have state income tax?
color	What colors make up a rainbow ?	NUMERIC	
creative	In what book can I find the story of Aladdin?	code	What is the telephone number for the University of Colorado?
currency	What currency is used in China?	count	About how many soldiers died in World War II?
disease/medicine	What does Salk vaccine prevent?	date	What is the date of Boxing Day?
event	What war involved the battle of Chapultepec?	distance	How long was Mao's 1930s Long March?
food	What kind of nuts are used in marzipan?	money	How much did a McDonald's hamburger cost in 1963?
instrument	What instrument does Max Roach play?	order	Where does Shanghai rank among world cities in population?
lang	What's the official language of Algeria?	other	What is the population of Mexico?
letter	What letter appears on the cold-water tap in Spain?	period	What was the average life expectancy during the Stone Age?
other	What is the name of King Arthur's sword?	percent	What fraction of a beaver's life is spent swimming?
plant	What are some fragrant white climbing roses?	temp	How hot should the oven be when making Peachy Oat Muffins?
product	What is the fastest computer?	speed	How fast must a spacecraft travel to escape Earth's gravity?
religion	What religion has the most members?	size	What is the size of Argentina?
sport	What was the name of the ball game played by the Mayans?	weight	How many pounds are there in a stone?
substance	What fuel do airplanes use?		
symbol	What is the chemical symbol for nitrogen?		
technique	What is the best way to remove wallpaper?		
term	How do you say "Grandma" in Irish?		
vehicle	What was the name of Captain Bligh's ship?		
word	What's the singular of dice?		

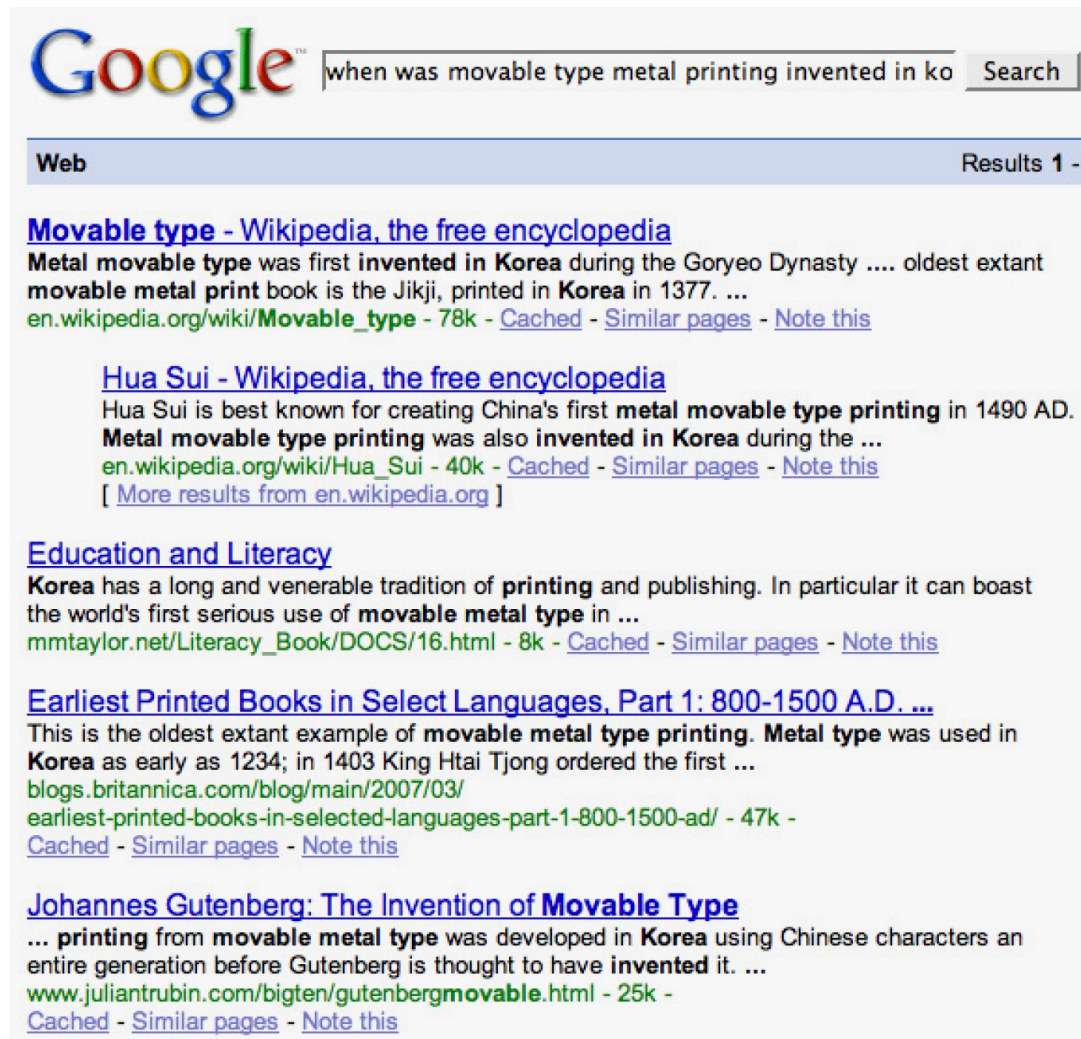
Answer Type Detection (Question Classification)

- **hand written rules:**
example for regular expression rule (assuming question has been (named-)entity-tagged):
who {is|was|are|were} PERSON → HUMAN:DESCRIPTION
- **machine-learning:**
 - rather **syntactic features**: words, POS-tags, (Named) Entity types of words,
 - rather **semantic features**: WordNet synset IDs of words, hypernyms of word, hyponyms of word
 - often most important feature: **question headword** (answer type word):
 - Which **city** in China has the largest number of foreign companies?
 - What is the state **flower** of California?

- from set of retrieved documents: detect possible **answer passages** (sections, paragraphs, sentences, ...)
 - **filter** out other passages
 - **rank** according to answer likelihood
- **filter**-step: run entity detection / answer type detection on all passages and filter out passages with non-matching types
- **ranking**-step: use supervised **classifier's score** on passage as rank; possible **features**:
 - number of named entities of the right type
 - number of question keywords
 - longest exact sequence of question keywords
 - rank of the corresponding document
 - proximity of keywords from query
 - N-gram overlap between the passage and the question

Passage Retrieval

- **passage extraction** using Google search: **use snippets**:
example: *when was movable type metal printing invented in Korea?*



The screenshot shows a Google search interface with the query "when was movable type metal printing invented in ko" entered in the search bar. The search button is labeled "Search". Below the search bar, the results are categorized under "Web" and show "Results 1 -".

Movable type - Wikipedia, the free encyclopedia
Metal movable type was first invented in Korea during the Goryeo Dynasty oldest extant movable metal print book is the Jikji, printed in Korea in 1377. ...
en.wikipedia.org/wiki/Movable_type - 78k - [Cached](#) - [Similar pages](#) - [Note this](#)

Hua Sui - Wikipedia, the free encyclopedia
Hua Sui is best known for creating China's first metal movable type printing in 1490 AD. Metal movable type printing was also invented in Korea during the ...
en.wikipedia.org/wiki/Hua_Sui - 40k - [Cached](#) - [Similar pages](#) - [Note this](#)
[[More results from en.wikipedia.org](#)]

Education and Literacy
Korea has a long and venerable tradition of printing and publishing. In particular it can boast the world's first serious use of movable metal type in ...
mmtaylor.net/Literacy_Book/DOCS/16.html - 8k - [Cached](#) - [Similar pages](#) - [Note this](#)

Earliest Printed Books in Select Languages, Part 1: 800-1500 A.D. ...
This is the oldest extant example of movable metal type printing. Metal type was used in Korea as early as 1234; in 1403 King Htai Tjong ordered the first ...
blogs.britannica.com/blog/main/2007/03/earliest-printed-books-in-selected-languages-part-1-800-1500-ad/ - 47k - [Cached](#) - [Similar pages](#) - [Note this](#)

Johannes Gutenberg: The Invention of Movable Type
... printing from movable metal type was developed in Korea using Chinese characters an entire generation before Gutenberg is thought to have invented it. ...
www.juliantrubin.com/bigten/gutenbergmovable.html - 25k - [Cached](#) - [Similar pages](#) - [Note this](#)

extract a **specific answer** from the passage:

- pattern extraction
- N-gram tiling
- **pattern extraction**: for all **entities** with the **right answer type** in high ranked passages use reg-ex rules to extract:
 - *who is the prime minister of India?*
Manmohan Singh [HUMAN], Prime Minister of India, had told left leaders that the deal would not be renegotiated.
 - *how tall is Mt. Everest?*
The official height of Mount Everest is 29029 feet [DISTANCE-QUANTITY]
- **particularly difficult**: REASON, DESCRIPTION, or DEFINITION type queries: → use special hand-written rules:
example: **rules for DEFINITION type** questions:

Pattern	Question	Answer
<AP> such as <QP>	What is autism?	“, <u>developmental disorders</u> such as autism”
<QP>, a <AP>	What is a caldera?	“the Long Valley caldera, a <u>volcanic crater</u> 19 miles long”

- pattern extraction: final step: **rank found candidate answers** by a supervised classifier's score using **features** such as:
 - **answer type match**
true if the candidate answer contains a phrase with the correct answer type.
 - **pattern match**
the identity of a pattern that matches the candidate answer.
 - **number of matched question keywords**
how many question keywords are contained in the candidate answer.
 - **keyword distance**
the distance between the candidate answer keywords and query keywords
 - **novelty factor**
true if at least one word in the candidate answer is not in the query.
 - **apposition features**
true if the candidate answer is an appositive to a phrase containing many question terms; appositive: example: *the fighter, a karateka, was attacking me with a straight kick*
 - **punctuation location**
true if the candidate answer is immediately followed by a comma, period, quotation marks, semicolon, or exclamation mark.
 - **sequences of question terms**
length of the longest sequence of question terms that occurs in the candidate answer.

- **N-gram tiling** (ide: exploit **redundancy** of information on the Web): use snippets returned from web-query (via reformulated question query):
3 steps:
 - **N-gram-mining**:
 - extract unigrams, bigrams and trigrams from snippet;
 - weight: function of N-gram frequency over snippets and of weight of query reformulation pattern
 - **N-gram filtering**:
 - score N-gram for match of predicted answer type
 - use hand-written filters for scoring
 - **N-gram tiling**:
 - concatenate overlapping N-gram fragments into longer answers
 - greedy approach: start with highest scored candidate, tile in other candidates, score resulting concatenation and add to candidate list and iterate

Neural Answer Extraction

- **idea**: compute question embedding + **embeddings** for candidate answer passages elements; compare via **similarity**
- Stanford Question Answering Dataset (**SQuAD** (2018)): passages from Wikipedia and associated questions (150000);
answers: spans from the passage; also: unanswerable questions

Beyoncé Giselle Knowles-Carter (born September 4, 1981) is an American singer, songwriter, record producer and actress. Born and raised in **Houston, Texas**, she performed in various **singing and dancing** competitions as a child, and rose to fame in the late 1990s as lead singer of R&B girl-group Destiny's Child. Managed by her father, Mathew Knowles, the group became one of the world's best-selling girl groups of all time. Their hiatus saw the release of Beyoncé's debut album, *Dangerously in Love* (**2003**), which established her as a solo artist worldwide, earned five Grammy Awards and featured the Billboard Hot 100 number-one singles "Crazy in Love" and "Baby Boy".

Q: "In what city and state did Beyoncé grow up?"

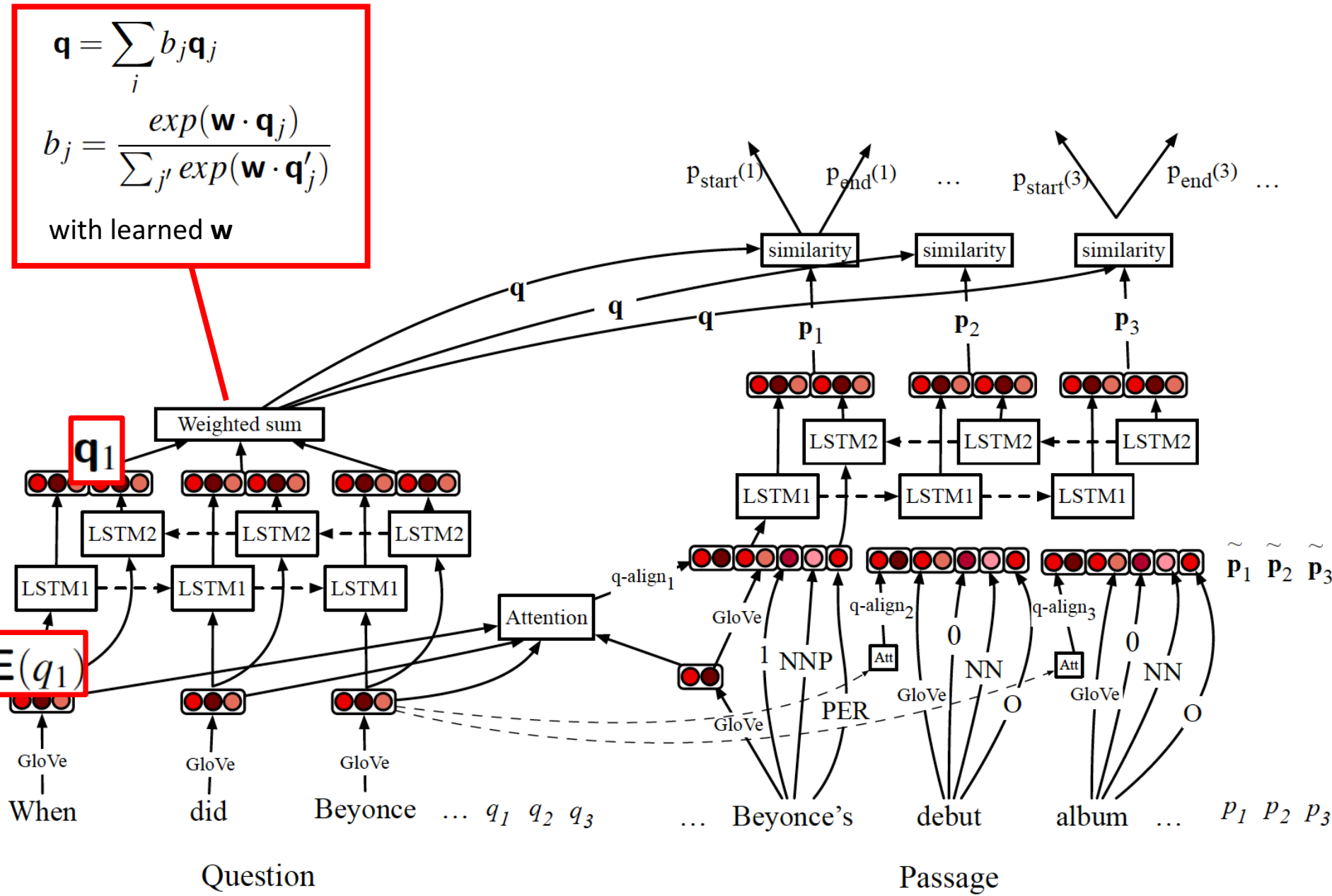
A: "**Houston, Texas**"

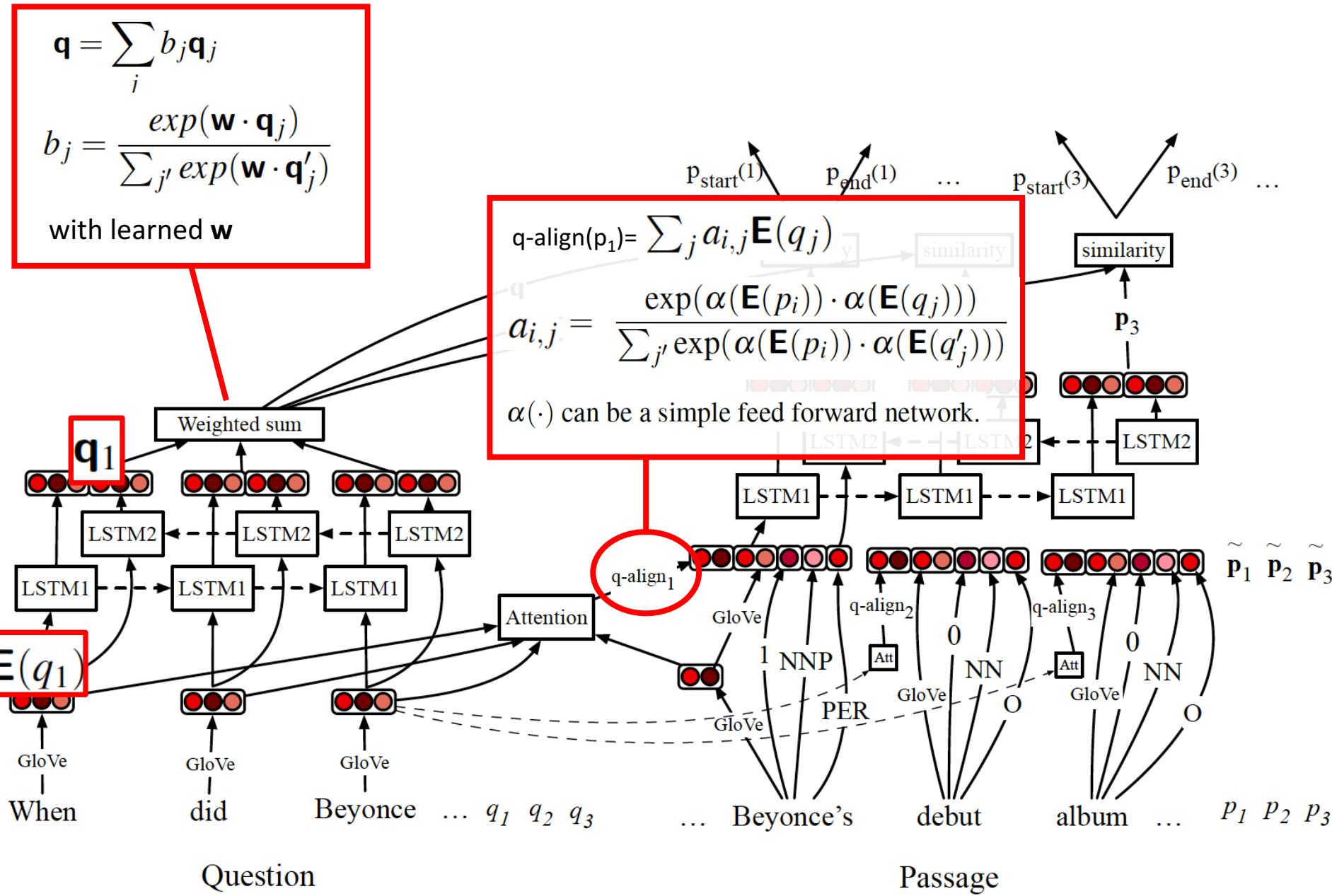
Q: "What areas did Beyoncé compete in when she was growing up?"

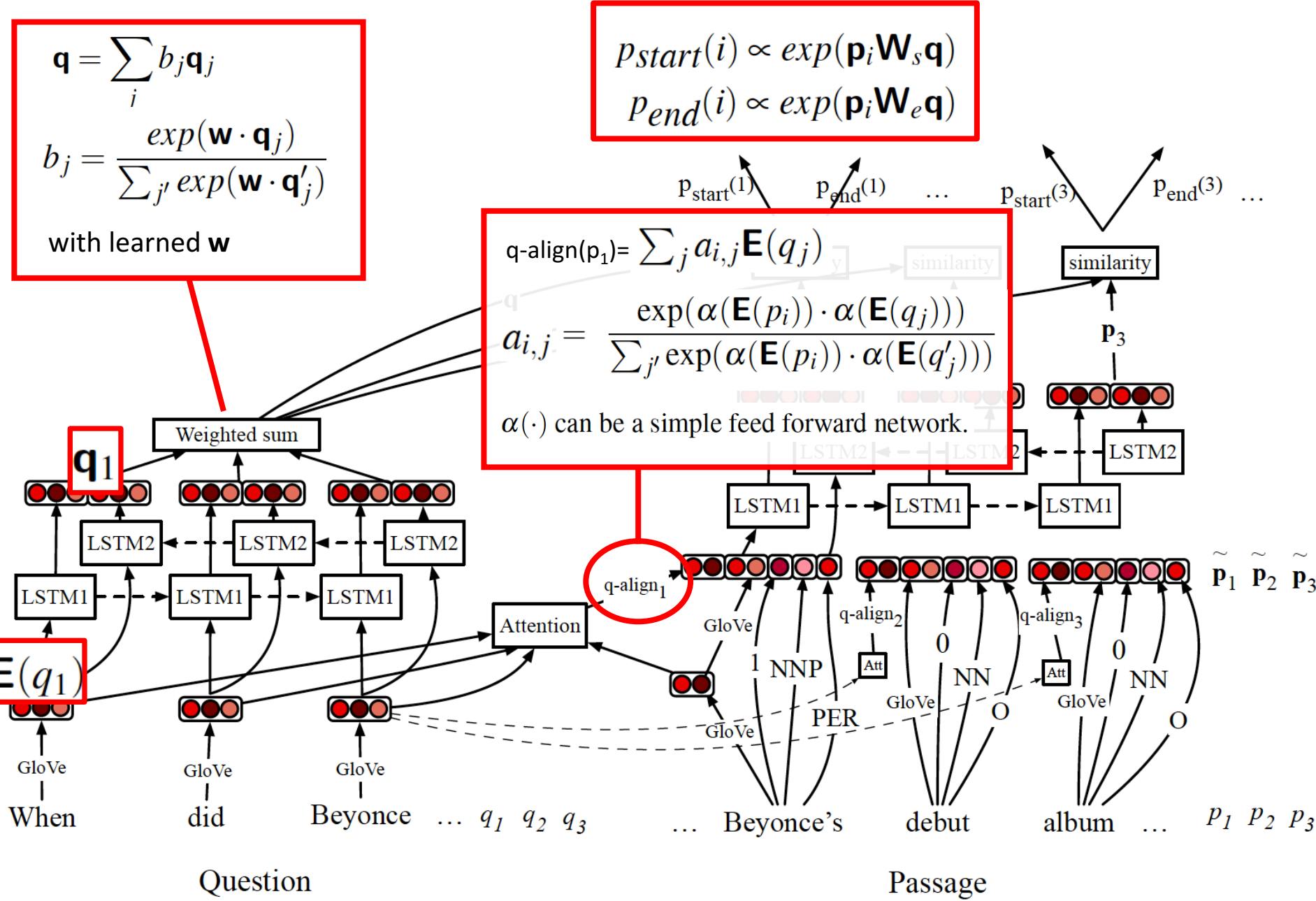
A: "**singing and dancing**"

Q: "When did Beyoncé release *Dangerously in Love*?"

A: "**2003**"







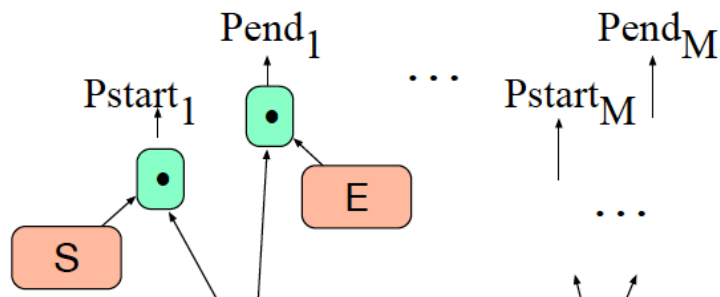
BERT-based Question Answering

$$Pstart_i = \frac{e^{S \cdot T_i'}}{\sum_j e^{S \cdot T_j'}}$$

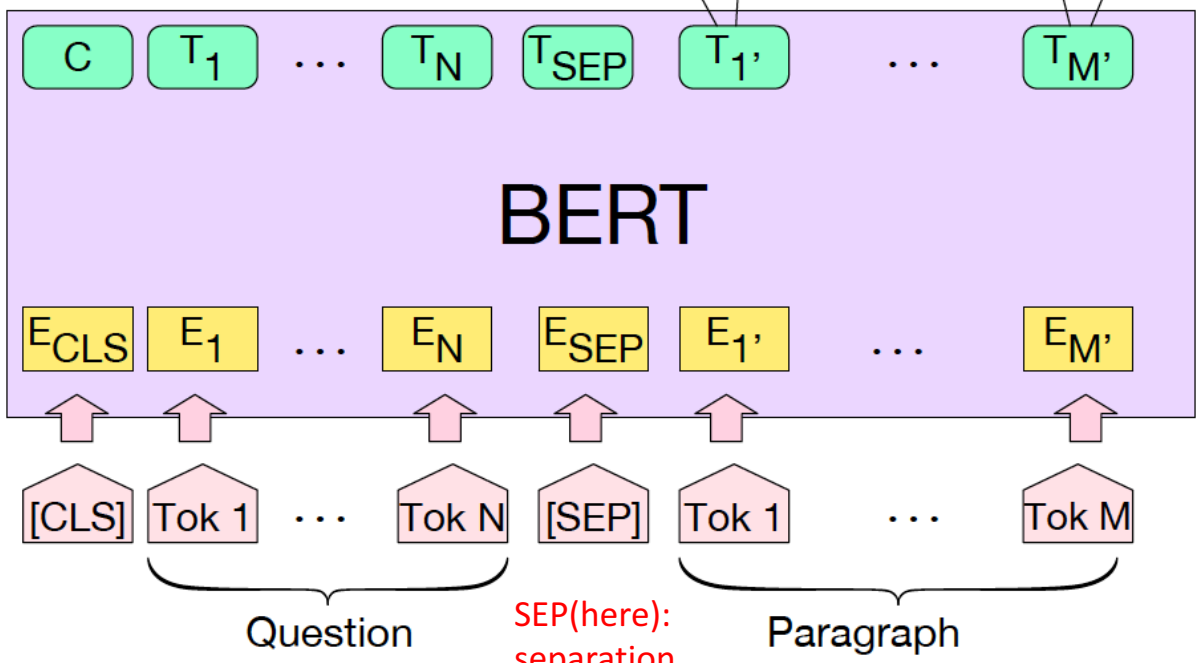
$$Pend_i = \frac{e^{E \cdot T_i'}}{\sum_j e^{E \cdot T_j'}}$$

score of a span
from i to j:
 $S \cdot T_i' + E \cdot T_j'$

S: trained general span-start embedding
E: trained general span-end embedding



CLS (here):
<start>



SEP(here):
separation
btw. question
and answer-containing paragraph

Knowledge-Based Question Answering

- answers in more structured database → map NL questions to structured queries (SQL (for relational DBs), SPARQL (for RDF repositories), 1st order logic (for general knowledge bases) etc.) (mapping: “**semantic parsing**”)

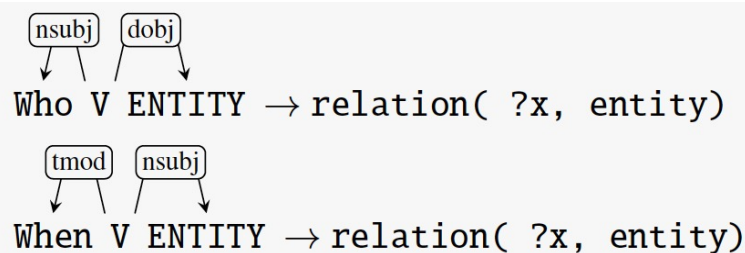
Question	Logical form
When was Ada Lovelace born?	<code>birth-year (Ada Lovelace, ?x)</code>
What states border Texas?	<code>λ x.state(x) ∧ borders(x,texas)</code>
What is the largest state	<code>argmax(λx.state(x), λx.size(x))</code>
How many people survived the sinking of the Titanic	<code>(count (!fb:event.disaster.survivors fb:en.sinking_of_the_titanic))</code>

- semantic parsing: **hand built rules** or via **supervised ML** 😊
- **hand built rules**: very similar to techniques introduced in section on relation extraction in chapter on information extraction

Rule-Based Approaches: Supervised Methods

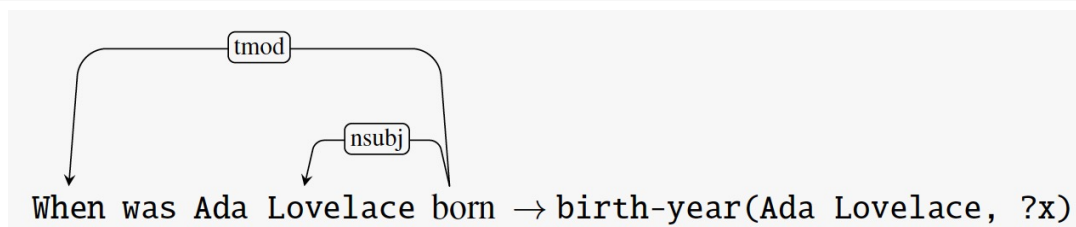
map **parse trees** to **logic form of query**: bootstrapping approach:

- start with lexicon (e.g. strings for all entities in database) and **simple general rules**

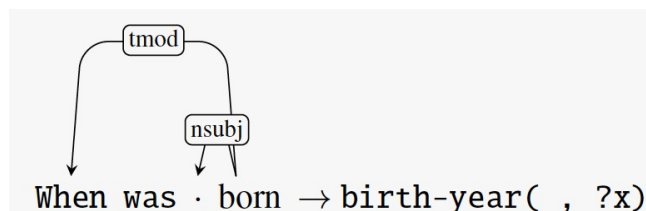


- **parse** training examples

“When was Ada Lovelace born?” → `birth-year (Ada Lovelace, ?x)`



and **induce** larger set of **more specific rules**, such as



Rule-Based Approaches: Supervised Methods

- bookkeep **counts** of rule applications in training data → derive **rule probabilities** for **disambiguation**
- inducing systems able to handle more complicated examples

What is the biggest state bordering Texas? →

$\text{argmax}(\lambda x. \text{state}(x) \wedge \text{borders}(x, \text{texas}), \lambda x. \text{size}(x))$

may need more complex default rules

Rule-Based Approaches: Semi-Supervised Methods

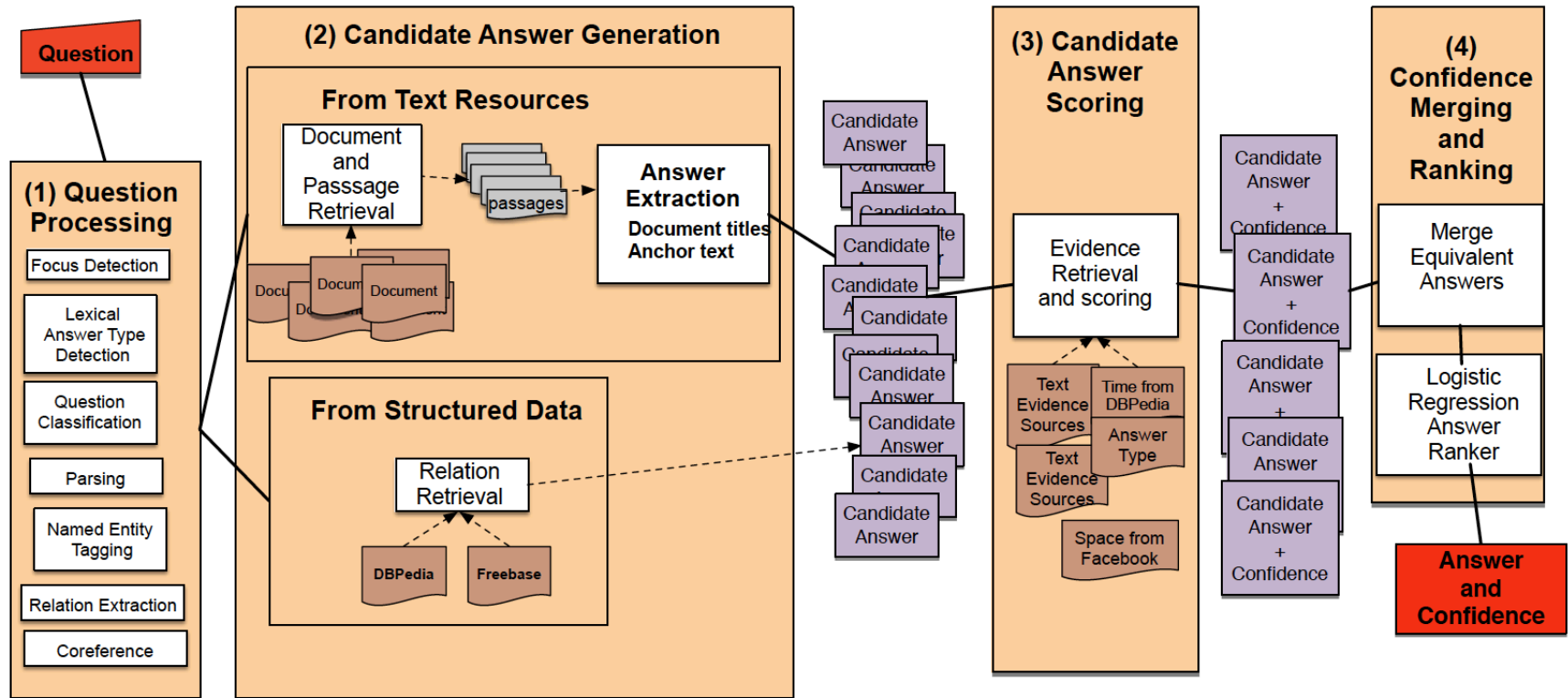
distant supervision:

- **example**: REVERB information extraction system: extracts billions of (subject, relation, object) triples of strings from the web, e.g. (“Ada Lovelace”, “was born in”, “1815”).
- now: normalize (e.g. times) and **align** (link) subject and object with structured knowledge base (e.g. Wikipedia) (**entity linking**)
- having aligned the entities and using relations in structured knowledge base (e.g. FreeBase entry: people.person.birthdate(ada lovelace,1815)): **align/link predicate to relation**: “was born in” \leftrightarrow people.person.birthdate

capital of	capital city of	become capital of
capitol of	national capital of	official capital of
political capital of	administrative capital of	beautiful capital of
capitol city of	remain capital of	make capital of
political center of	bustling capital of	capital city in
cosmopolitan capital of	move its capital to	modern capital of
federal capital of	beautiful capital city of	administrative capital city of

example phrases aligning with Freebase relation country.capital

Example: IBM Watson



- won the Jeopardy tv quiz show in 2011

Example: IBM Watson: Question Processing

question processing with DeepQA system:

- parsing, named entity tagging, and relation extraction.
- then extracts **focus** , **answer type** (also called the lexical answer type or LAT), and performs **question classification** and **question sectioning**.

example Jeopardy questions:

- POETS AND POETRY: ***He** was a bank clerk in the Yukon before he published “Songs of a Sourdough” in 1907.*
- THEATRE: *A new play based on this **Sir Arthur Conan Doyle** canine **classic** opened on the London stage in 2007.*
- **NE tagging** → (Sir Arthur Conan Doyle: PERSON, Yukon: GEOPOLITICAL ENTITY, “Songs of a Sourdough”: COMPOSITION)
- **co-reference resolution**: *he* ↔ *bank clerk*
- **relation extraction**:
authorof(focus, “Songs of a sourdough”), in (e2, e1, 1907),
publish (e1, he, “Songs of a sourdough”), temporallink(publish(...), 1907)

Example: IBM Watson: Question Processing

- **extraction of focus** (entity that co-refers with answer) (Q1: *he*, Q2: *Sir Arthur Conan Doyle canine classic*): **hand written rules** (\leftrightarrow stylized Jeopardy question format) (e.g. extract NP with DET *this*)
- **extraction of lexical semantic answer type**: words which tell something about the semantic type of the answer (*he*, *classic*) :
IBM analysis of Jeopardy: 20000 questions, 5000 (!) answer types (NE classes): with **rules**:
 - default rule: choose syntactic headword of the focus
 - possible additional lexical answer types \leftrightarrow words co-referent with focus or in particular syntactic relation with focus (e.g. headwords of appositives or predicative nominatives of the focus)
 - also possible: use found matching rules / answer types as features for ML classifier
 - difference btw. DeepQA and purely IR-based factoid question answerers described on slide 3ff: DeepQA generates lots of possible (answer, answer type) tuples and selects via candidate answer scoring

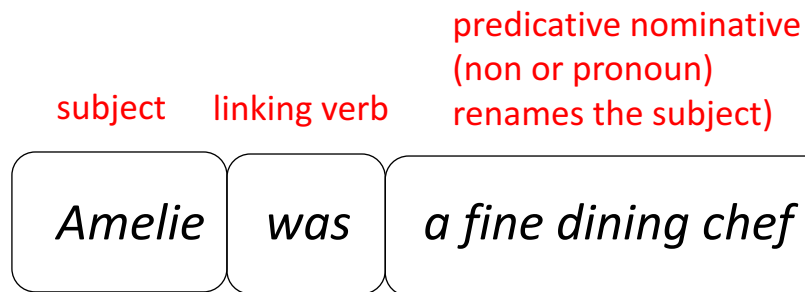
Example: IBM Watson: Question Processing

- extraction

Conan Doyle
question

- extraction

about the
IBM anal
classes):



[2]

- default rule: choose syntactic headword of the focus
- possible additional lexical answer types \leftrightarrow words co-referent with focus or in particular syntactic relation with focus (e.g. headwords of appositives or **predicative nominatives** of the focus)
- also possible: use found matching rules / answer types as features for ML classifier
- difference btw. DeepQA and purely IR-based factoid question answerers described on slide 3ff: DeepQA generates lots of possible (answer, answer type) tuples and selects via candidate answer scoring

Example: IBM Watson: Candidate Answer Generation

- answer candidates from **structured resources** (DBpedia, IMDB, FreeBase, DBPedia, etc.): create query from processed question (use extracted relation):
 - `authorof(focus,"Songs of a sourdough") → authorof(?x,"Songs of a sourdough")`
- answer candidates via **information retrieval**: use passage retrieval methods (slide 10f): DeepQA: stop-word removal + upweighting terms which occur in any relation with the focus:
 - *MOVIE: Robert Redford and Paul Newman starred in this depression era grifter flick* →
(2.0 Robert Redford) (2.0 Paul Newman) star depression era grifter (1.5 flick)
 - in retrieved passages: use heuristics: e.g.
 - anchor text (between <a> tags)
 - all NPs that are Wikipedia article headings

Example: IBM Watson: Candidate Answer Scoring

- example 1: **ontology based** answer scoring:
 - (candidate answer, answer type): score match between candidate answer and answer type:
 - (“difficulty swallowing”, “manifestation”): DeepQA: match words with entities in ontologies like DBpedia and WordNet:
“difficulty swallowing” \leftrightarrow DBpedia entity “Dysphagia” \leftrightarrow WordNet type “Symptom”;
“manifestation” \leftrightarrow WordNet type “Condition”.
 \rightarrow hyponymy, instance-of, or synonymy btw. “Symptom” and “Condition”? \rightarrow yes, hyponymy found in WordNet \rightarrow high score
- example 2: **passage retrieval based** answer scoring:
 - compare overlap between (a) passages retrieved by original question
(b) modified question where question focus is replaced with candidate answer \rightarrow score

Example: IBM Watson: Answer Merging and Scoring

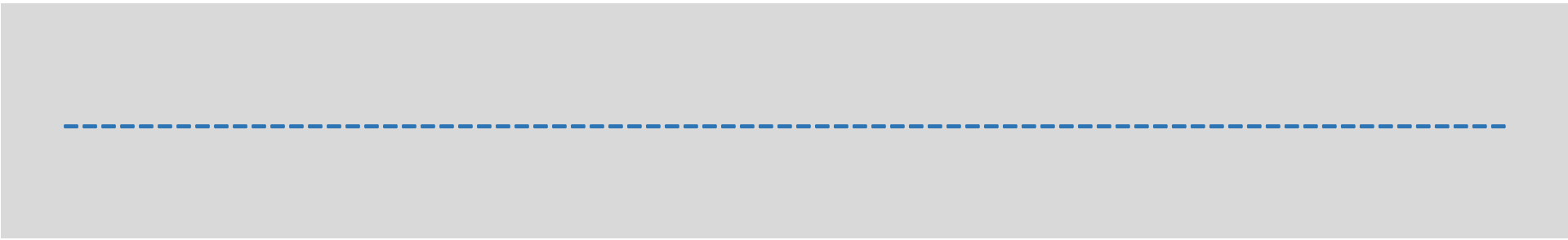
- what we have now: **answer candidates plus score vectors** as features
- first step: unify all **equivalent answers**, combining score vectors
 - using large automatically created **synonym dictionaries** of anchor text strings that point to the same Wikipedia page (example: *JFK, John F. Kennedy, John Fitzgerald Kennedy, Senator John F. Kennedy, President Kennedy, Jack Kennedy*)
 - for nouns: use **morphological parsers**, identifying morphological variants
- second step: use trained **classifier** (classes: “correct”, “incorrect”; training data: large set of all kinds of on answer candidates’ features) to produce **confidence values** (probabilities for class correct) for answer candidates of previous step;
- **rank** answer candidates according to confidence, iterate steps 1 and 2

Evaluation of Factoid Answers

- **Mean Reciprocal Rank** (MRR): corpus of questions plus correct answers; system delivers ranked list of possible answers: score == reciprocal rank of first correct answer. → for N questions:

$$\text{MRR} = \frac{1}{N} \sum_{i=1 \text{ s.t. } \text{rank}_i \neq 0}^N \frac{1}{\text{rank}_i}$$

- or **F1 score**: predicted answer and gold std answer as **sets of tokens**
- **corpora**:
 - TrecQA (2004),
 - Free917 (917 question-answer pairs) (2013),
 - Webquestions (5810 question-answer pairs) (2013)
 - SQuAD (2016)
 - NarrativeQA (2018)



- (1) Dan Jurafsky and James Martin: Speech and Language Processing (3rd ed. draft, version Oct 2019); Online: <https://web.stanford.edu/~jurafsky/slp3/> (URL, Oct 2019); this slide-set is especially based on chapter 25
- (2) see <http://grammar-monster.com> (URL, Oct 2019)

Recommendations for Studying

- minimal approach:

work with the slides and understand their contents! Think beyond instead of merely memorizing the contents

- standard approach:

minimal approach + read the corresponding pages in Jurafsky [1]

- interested students

== standard approach