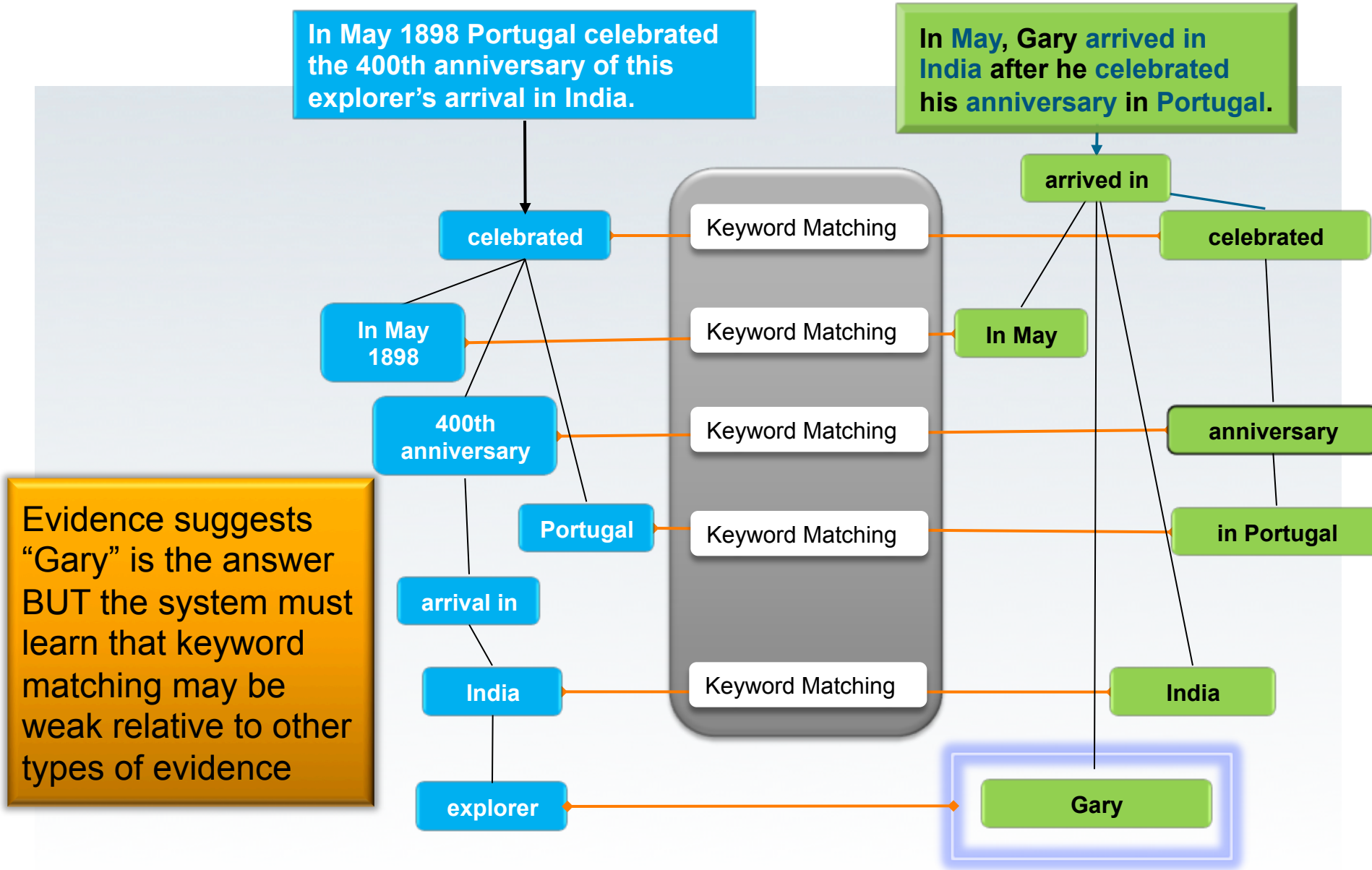
Semantic Technologies in IBM Watson™

Lesson 4 – Natural Language Processing

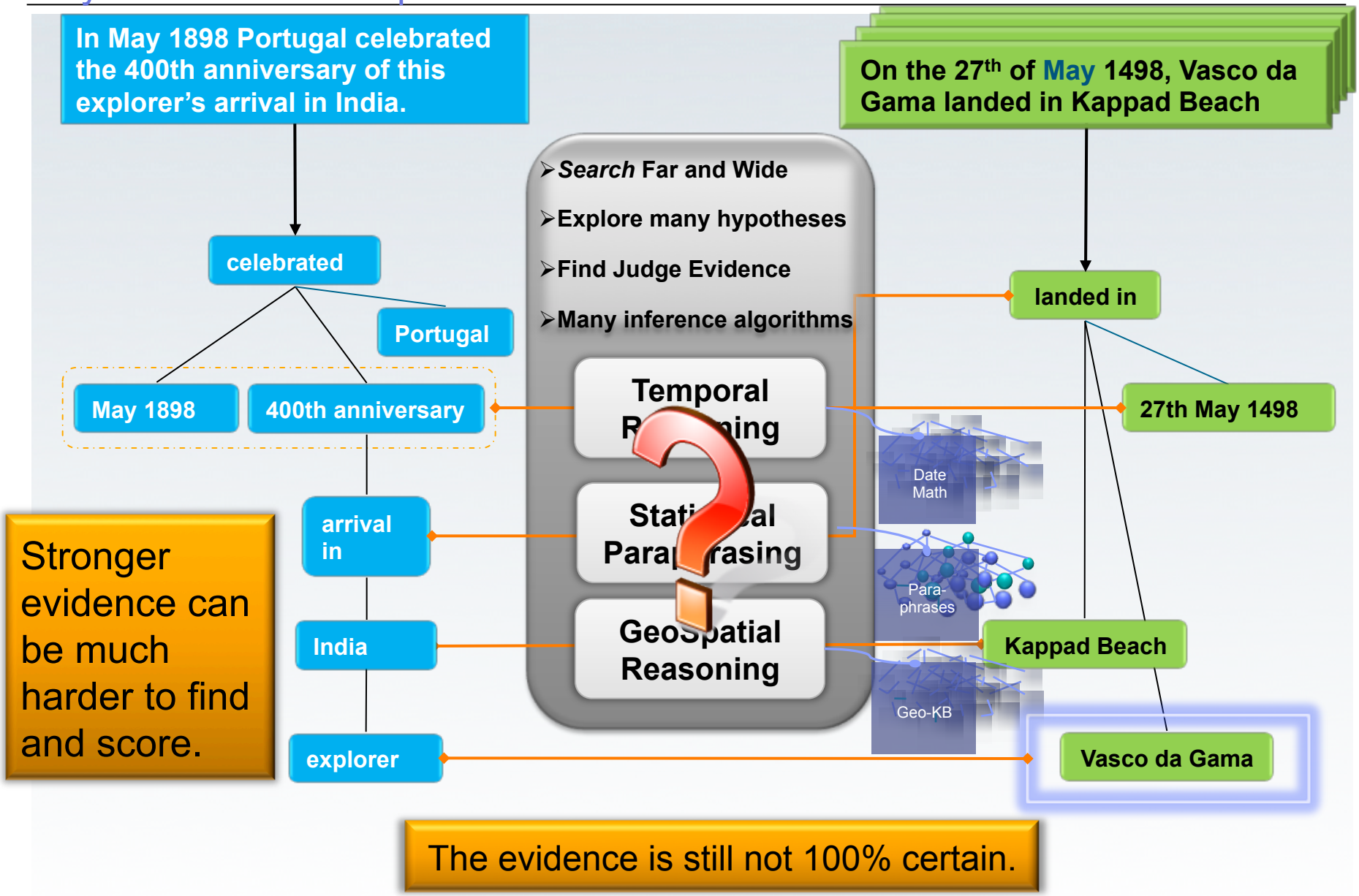
Professor: Alfio Massimiliano Gliozzo
TA: Or Biran

Keyword Evidence



Evidence suggests "Gary" is the answer BUT the system must learn that keyword matching may be weak relative to other types of evidence

Why Semantics? Deeper Evidence



Outline

- The NLP Stack
- Question Analysis
- Passage Scoring

Outline

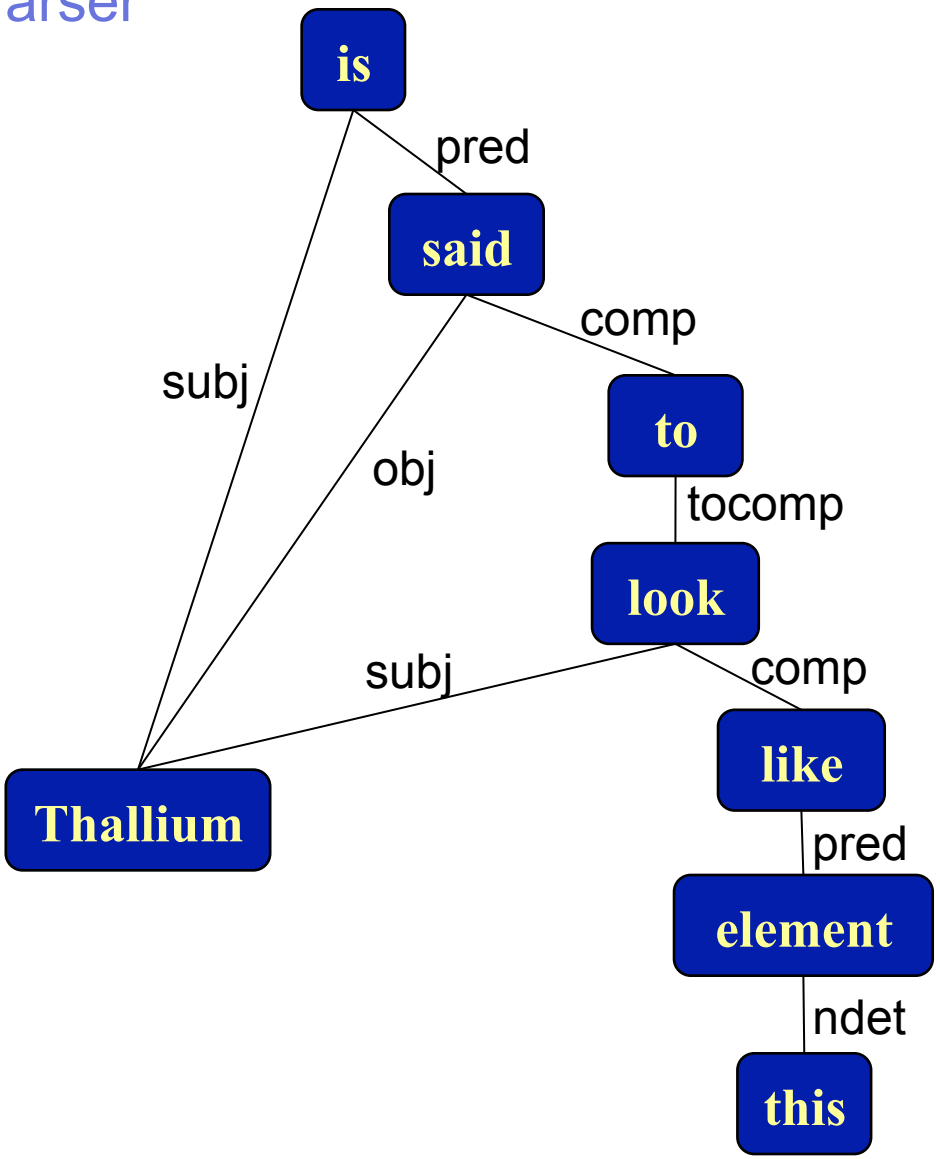
- The NLP Stack
- Question Analysis
- Passage Scoring

NLP Stack

- ESG: English Slot Grammar Parser
- Predicate Argument Structure (simplified parse)
- Rule-Based Named Entity Detection
- Intra-Paragraph Anaphora Resolution
- Temporal Normalization
- Temporal Arithmetic
- Pattern-Based Semantic Relation Detection
- Statistical Semantic Relation Detection

ESG: English Slot Grammar Parser

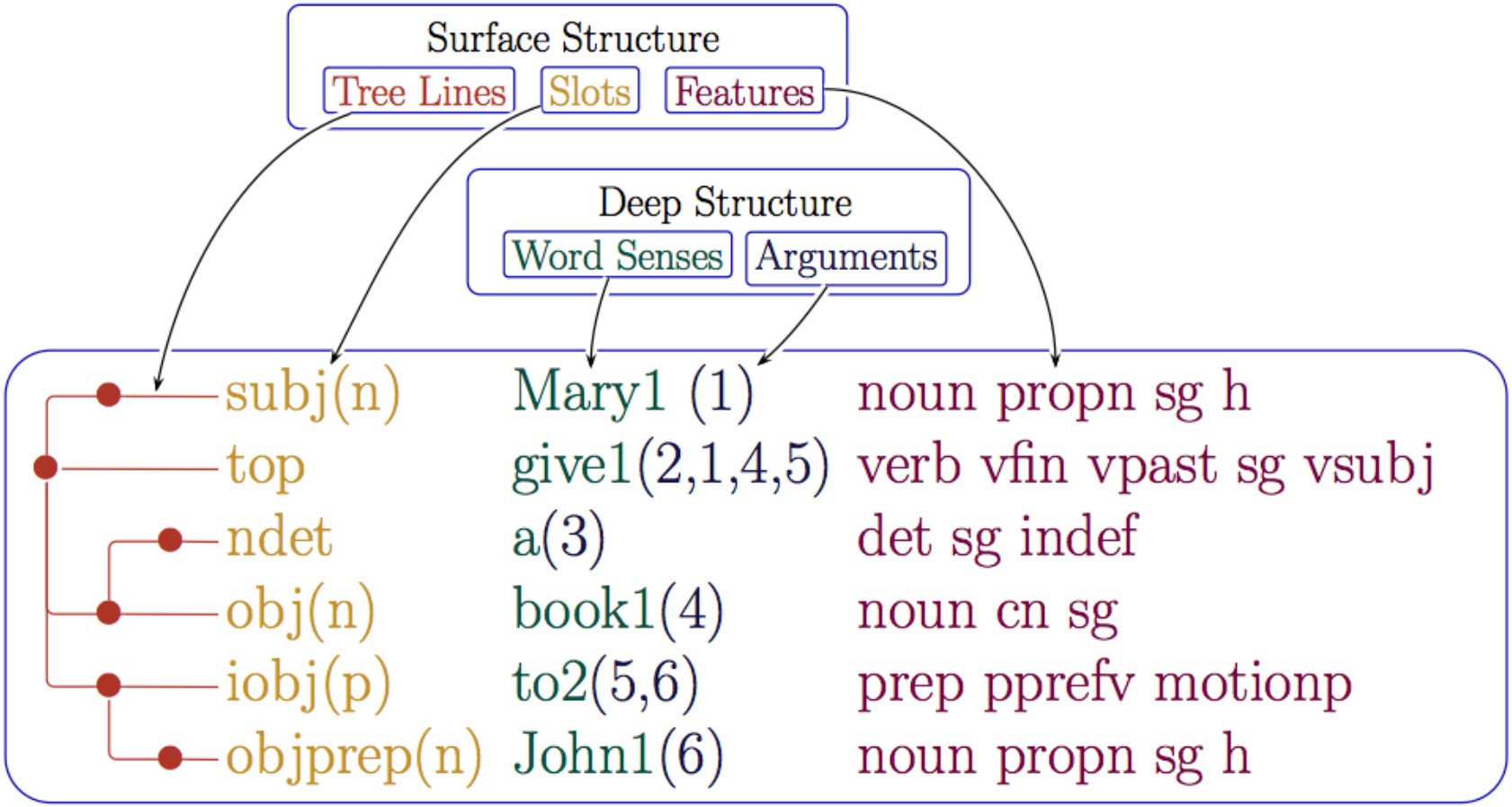
Thallium is said to look like this element



Extended Slot Grammar (ESG) Parser

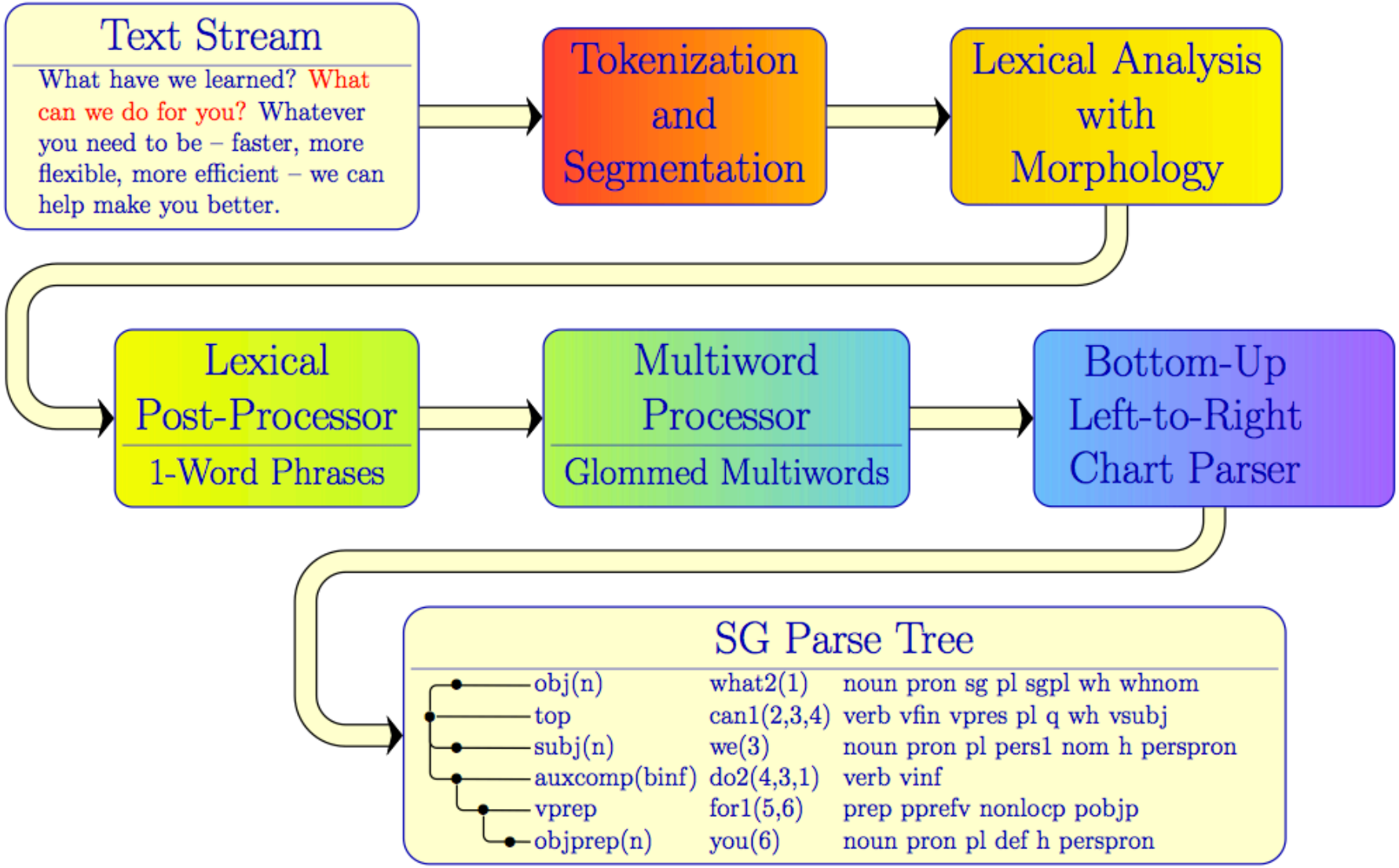
- The SG parser is a bottom-up, left-to-right chart parser.
- Rule-based (not statistical) – although it does use numerical scoring to arrive at most likely parses.
- SG can use multiple lexicons. There is always a main (base) lexicon, and there can be addendum lexicons
  - Easy domain adaptation
- The lexicons are indexed by citation forms (lemmas) of words. Morpholexical analysis.
- ESG base lexicon has about 88,000 entries, but many more word forms are recognized because of morphology and addendum lexicons.

Standard Slot Grammar Parse Display



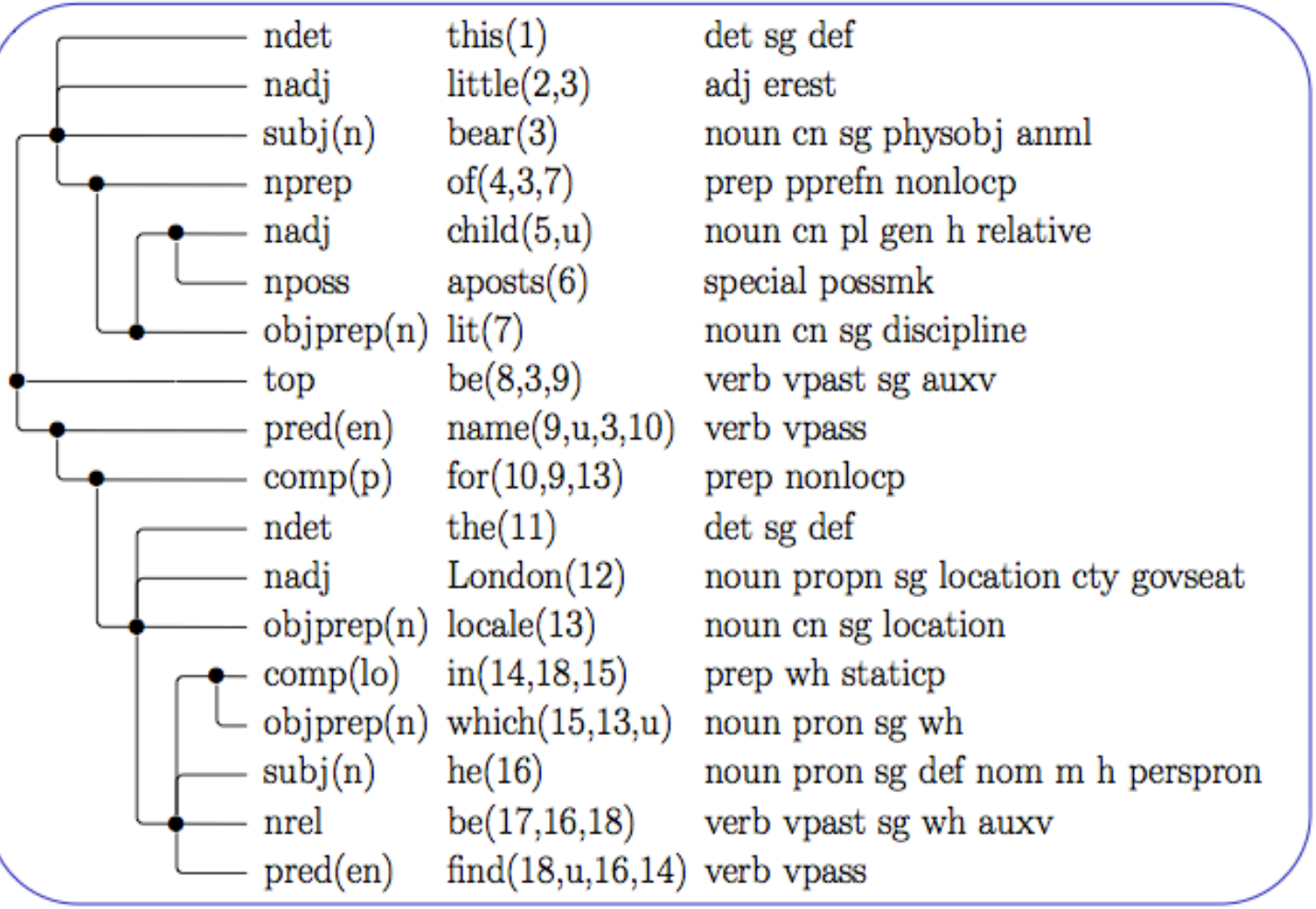
Mary gave a book to John.

Pipeline for SG Parsing



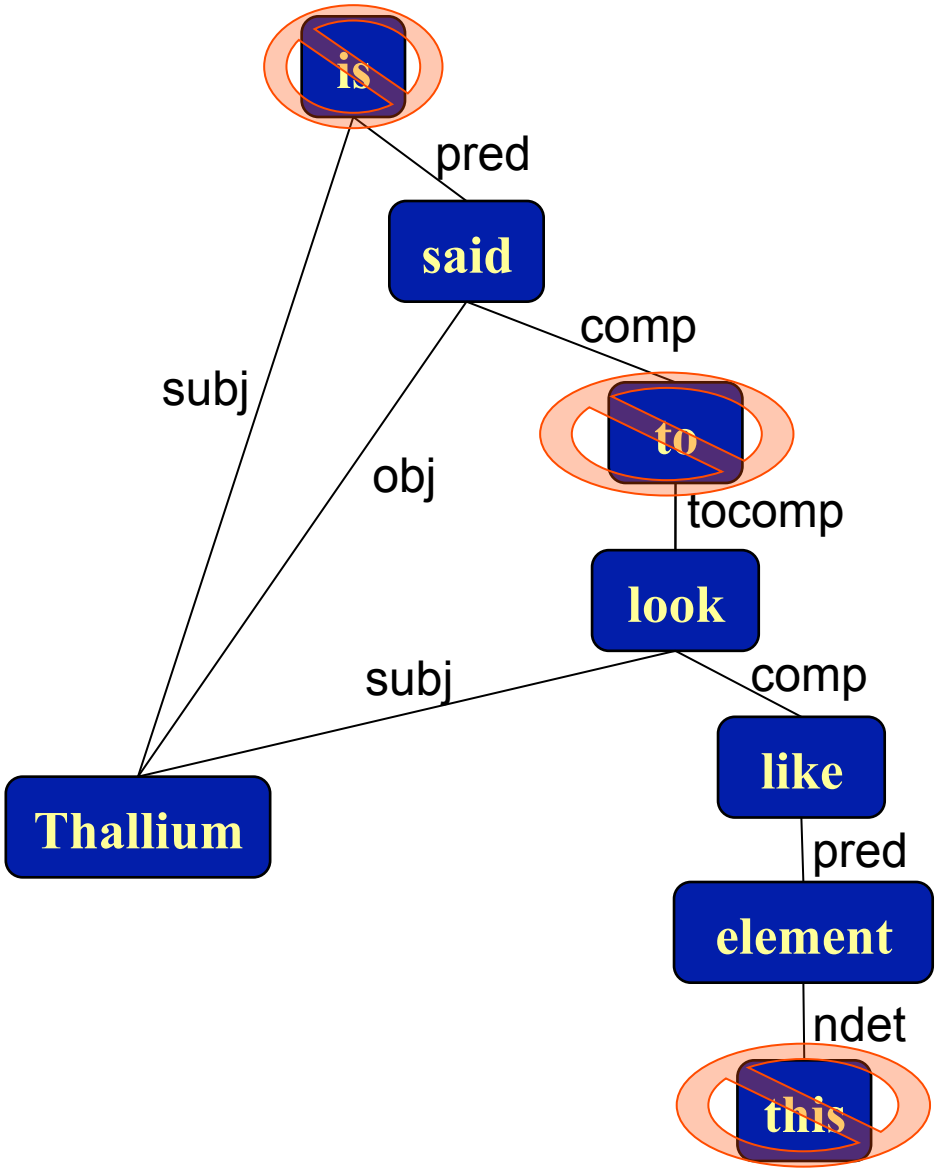
Parse of Jeopardy! Clue

This little bear of children's lit was named for the London locale in which he was found.



Predicate Argument Structure

Thallium is said to look like this element



Predicate Argument Structure: A layer of abstraction

Simplifies the ESG parse

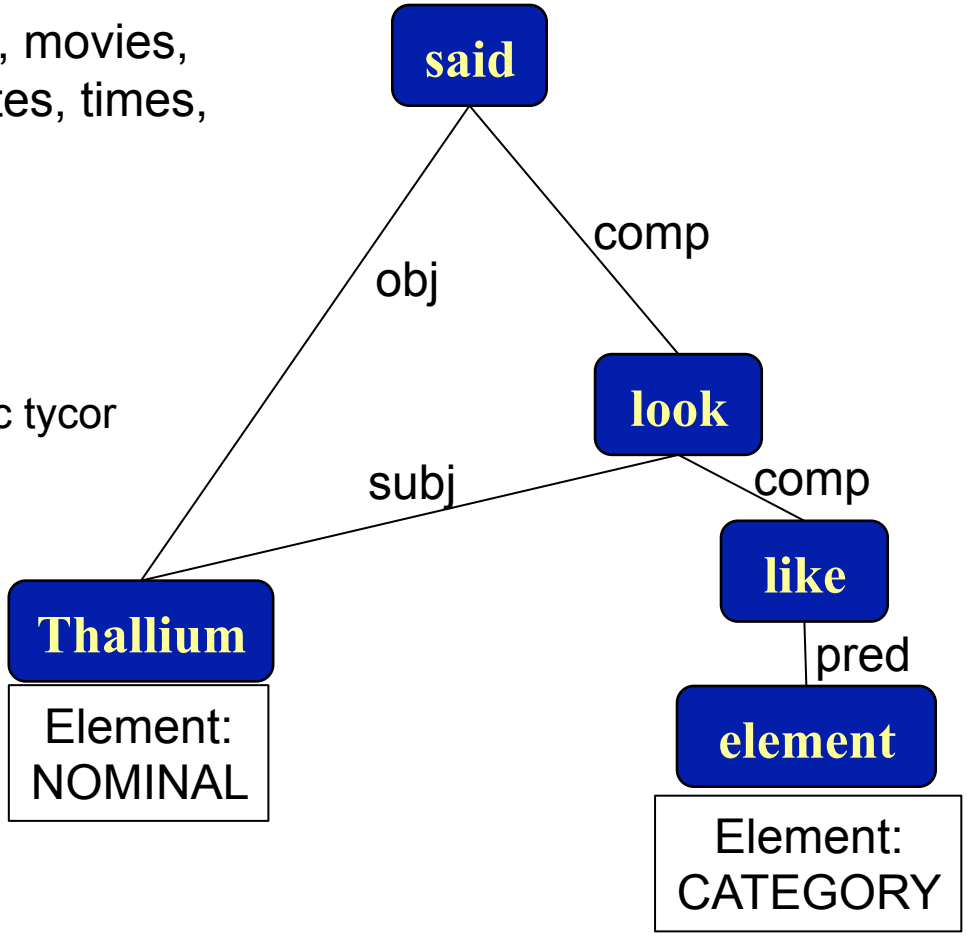
Does not add any new analysis

- Removes “unnecessary” nodes
- Simpler set of parts-of-speech
- Simplifies the encoding of lemma forms (no derivational morphology markers)
- Single dimension of links among parse nodes (no distinction between “deep” and “surface” structure)
- Both ESG and PAS have the usual suspects:
  - **noun**, **verb**, **adj** (adjective), **adv** (adverb), **prep** (preposition)
- ESG parse annotations have an uninflectedWord feature that corresponds roughly to the PAS lemmaForm feature

Named-Entity Detection

- Hundreds of entity types
- E.g., people, nations, cities, books, movies, weapons, musical instruments, dates, times, food, tools, elements
- Used in
  - Relation Detection
  - Answer lookup
  - Answer typing (positive in a specific tycor component)

Thallium is said to look like this element



- Extensible, Domain-Independent Named Entity Recognizer
- UIMA-based
  - Is itself a multi-annotator aggregate
- Primarily targets HUTT type system
  - About 400 types, about 170 of which in R2
- Recognition by lists, patterns, or both
  - Simple pattern file syntax, somewhat like regex

Relation Extraction

- Relation extraction: to classify the relation between two entity mentions into one of predefined relation classes `locatedAt? customerOf? employedBy?`
- Example:
  - "***The New Jersey Devils*** have signed ***Adam Larsson*** to a three-year, entry-level contract"
- Applications:
  - Information extraction
  - Database population
  - Machine reading
  - Question answering
  - Etc … …
- Challenges
  - Ambiguity: *Thomas Jefferson has signed the Declaration of Independence*
  - Context: … Nicole Kidman (1967)… vs. … Nicole Kidman (1990)…
  - Expressiveness of language:
    - IBM hired James, James started at IBM, James of IBM, …

Rule Based

Prolog Rules

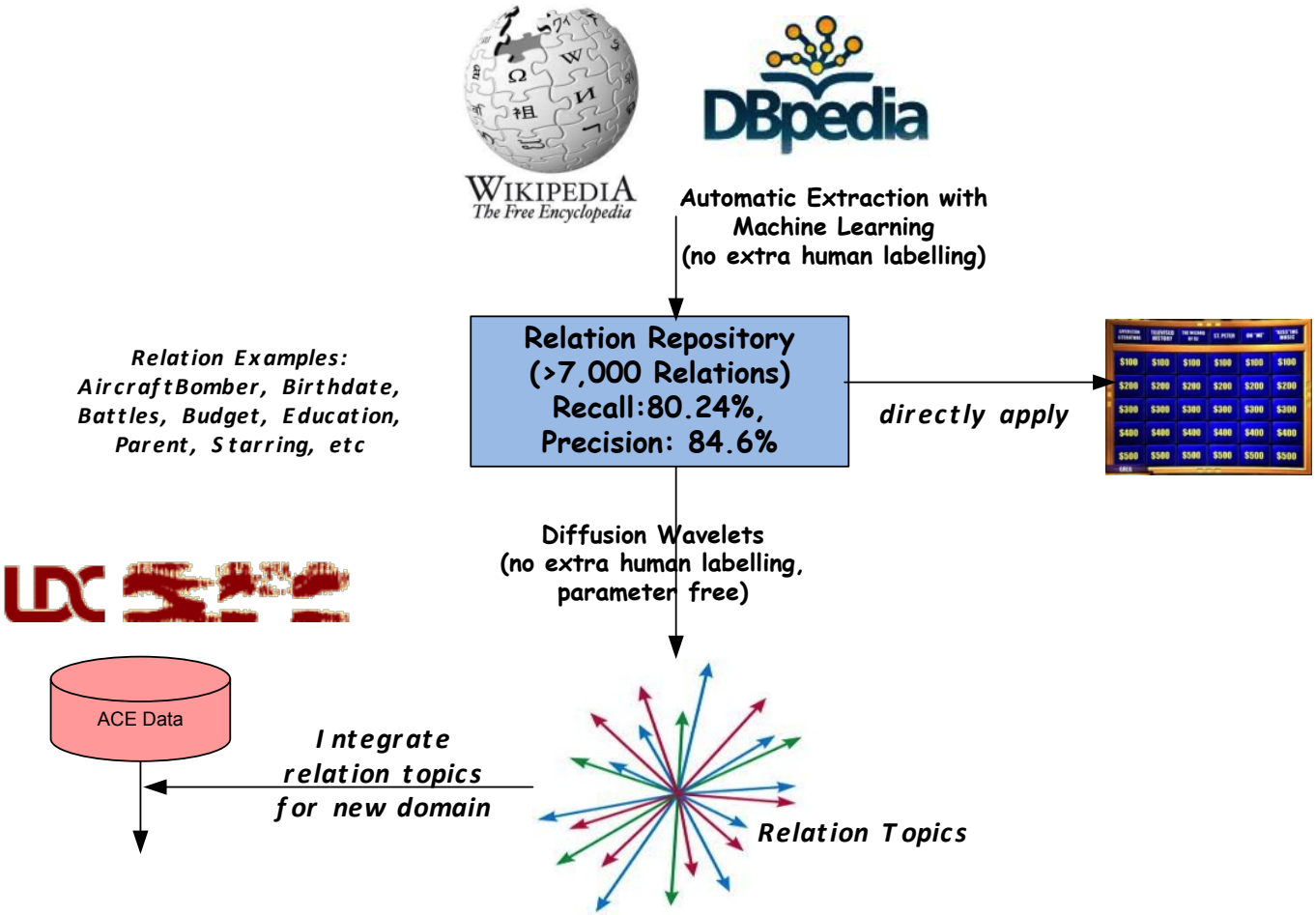
```
authorOf(Author,Composition) :-
  createVerb(Verb),
  subject(Verb,Author),
  author(Author),
  object(Verb,Composition),
  composition(Composition).

createVerb(Verb) :-
  partOfSpeech(Verb,verb),
  lemma(Verb,VerbLemma),
  member(VerbLemma,["write","publish'
```

Inferred

```
author(1)
composition(3)
authorOf(1,3)
```

TWREX - Topicalized Wide Relation and Entity eXtraction

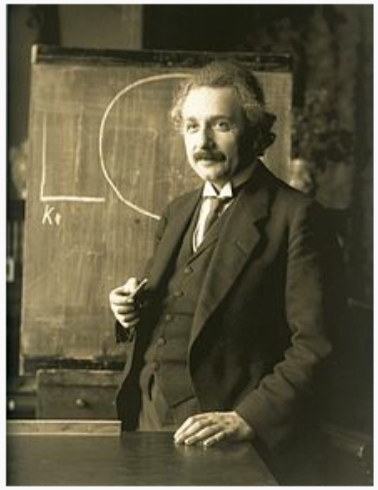


RELATION EXTRACTION

Distant supervision in TWREX

- Collecting training example from Wikipedia
- First occurrence heuristic on Dbpedia relations in Wikipedia,
  - For example, the Wikipedia page for “Albert Einstein” contains an infobox property “alma mater” with value “University of Zurich”, and the first sentence mentioning the arguments is the following: “Einstein was awarded a PhD by the University of Zurich”, which expresses the relation.

Albert Einstein



Albert Einstein in 1921

Born	14 March 1879 Ulm, Kingdom of Württemberg, German Empire
Died	18 April 1955 (aged 76) Princeton, New Jersey, United States
Residence	Germany, Italy, Switzerland, Austria, Belgium, United Kingdom, United States
Citizenship	Kingdom of Württemberg (1879–1896) Stateless (1896–1901) Switzerland (1901–1955) Austria–Hungary (1911–1912) German Empire (1914–1933) United States (1940–1955)

Early life and education

See also: Einstein family

Albert Einstein was born in Ulm, in the Kingdom of Württemberg in the German Empire on 14 March 1879.[10] His father was Hermann Einstein, a salesman and engineer. His mother was Pauline Einstein (née Koch). In 1880, the family moved to Munich, where his father and his uncle founded *Elektrotechnische Fabrik J. Einstein & Cie*, a company that manufactured electrical equipment based on direct current.[10]

R2 Single-Word-Pattern Annotator

- Used when tokens have internal structure, so not enumerable
  - E.g. “5ft”
- Driven by a SW pattern file
- Regular-Expression based
- Only triggered if numerals occur in token
- Pattern-match causes R2 type-system annotation to occur, for recognition later
- User-extensible without type system extension or recompilation/ rebuilding

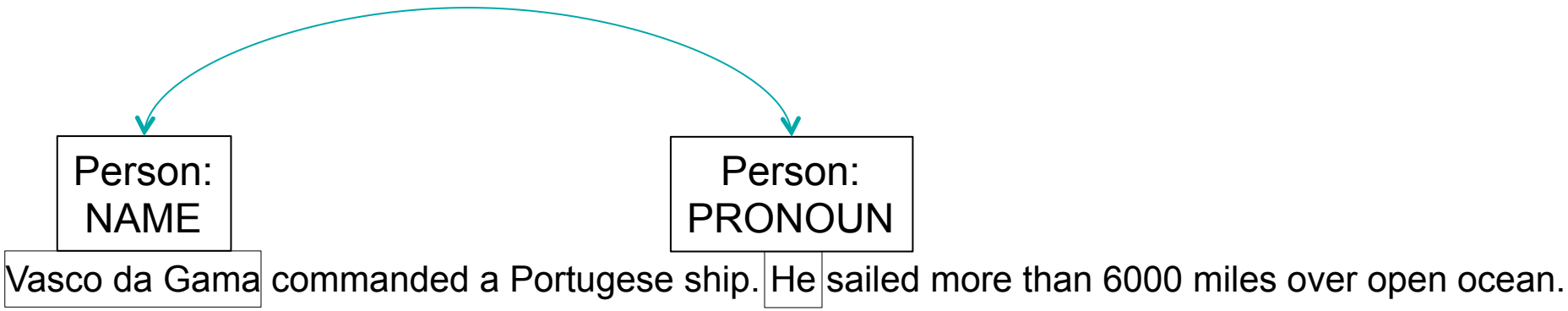
E.g. for recognizing areal expressions

```
\d{1,3}-?acre                      AREA   4-acre
\d{1,3}\,\d\d\d-?acre              AREA   4,123-acre
\d{1,3}\,\d\d\d,\d\d\d-?acre       AREA   4,123,987-acre
\d{1,3}\.\d{1,3}-?acre             AREA   43.987-acre
```

- The identifier “**:AREA**” can be used in R2 pattern files

Intra-Paragraph Anaphora Resolution (IPAR)

- Used in
  - Question Analysis to identify the Focus (and LATs)
  - Passage Scoring (LFACS, Skip Bigram)

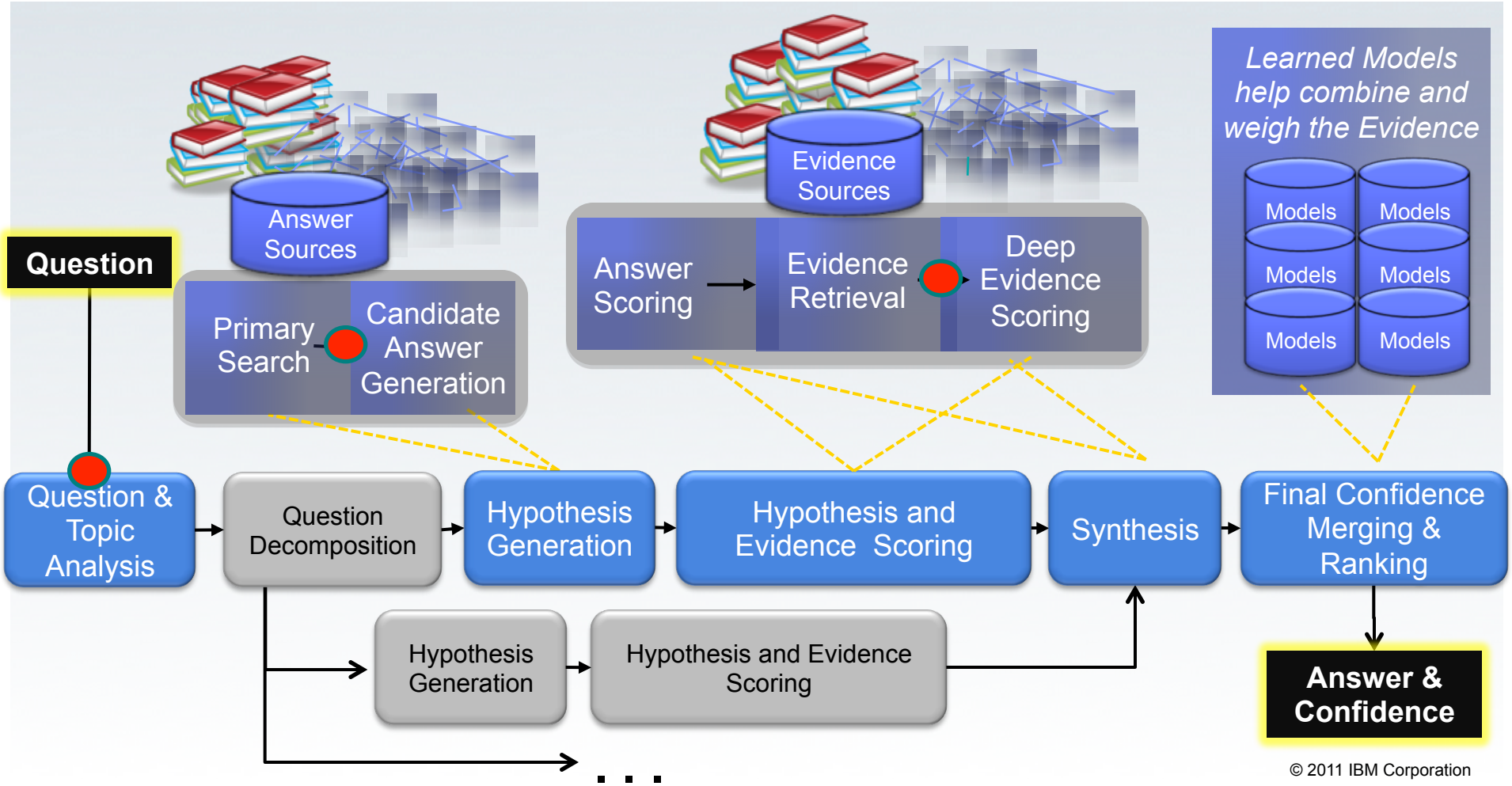


IPAR Operation

- Creates Co-reference chains between entities with compatible Hutt annotations
- Moves left-to-right within paragraph, at each Hutt entity (source) checks for compatibility with any other Hutt entity (target) to its left.
- Co-reference link made iff:
  - Type of target <= type of source
  - Gender of target = gender of source (if both known)
  - Number of target = number of source (if both known)
  - …

Where is the NLP stack run in the DeepQA pipeline?

- On questions, at the start of question analysis
- On primary search results, before candidate answer generation
- On supporting evidence, before deep evidence scoring



Outline

- The NLP Stack
- Question Analysis
- Passage Scoring

Question Analysis: what?

- *POETS & POETRY: He was a bank clerk in the Yukon before he published "Songs of a Sourdough" in 1907*
- *Identify*
  - *Focus:* part of the question that is a reference to the answer
    - *E.g. He*
  - *Lexical Answer Types:* terms in the question that indicate what type of entity is being asked for
    - "He," "clerk," and "poet"
  - *Question Classification:* Factoid (Most Jeopardy Questions), Definition, Multiple-Choice, Puzzle, Common-Bonds, Fill-in-the-blanks, and Abbreviation.
    - *Factoid*

Focus

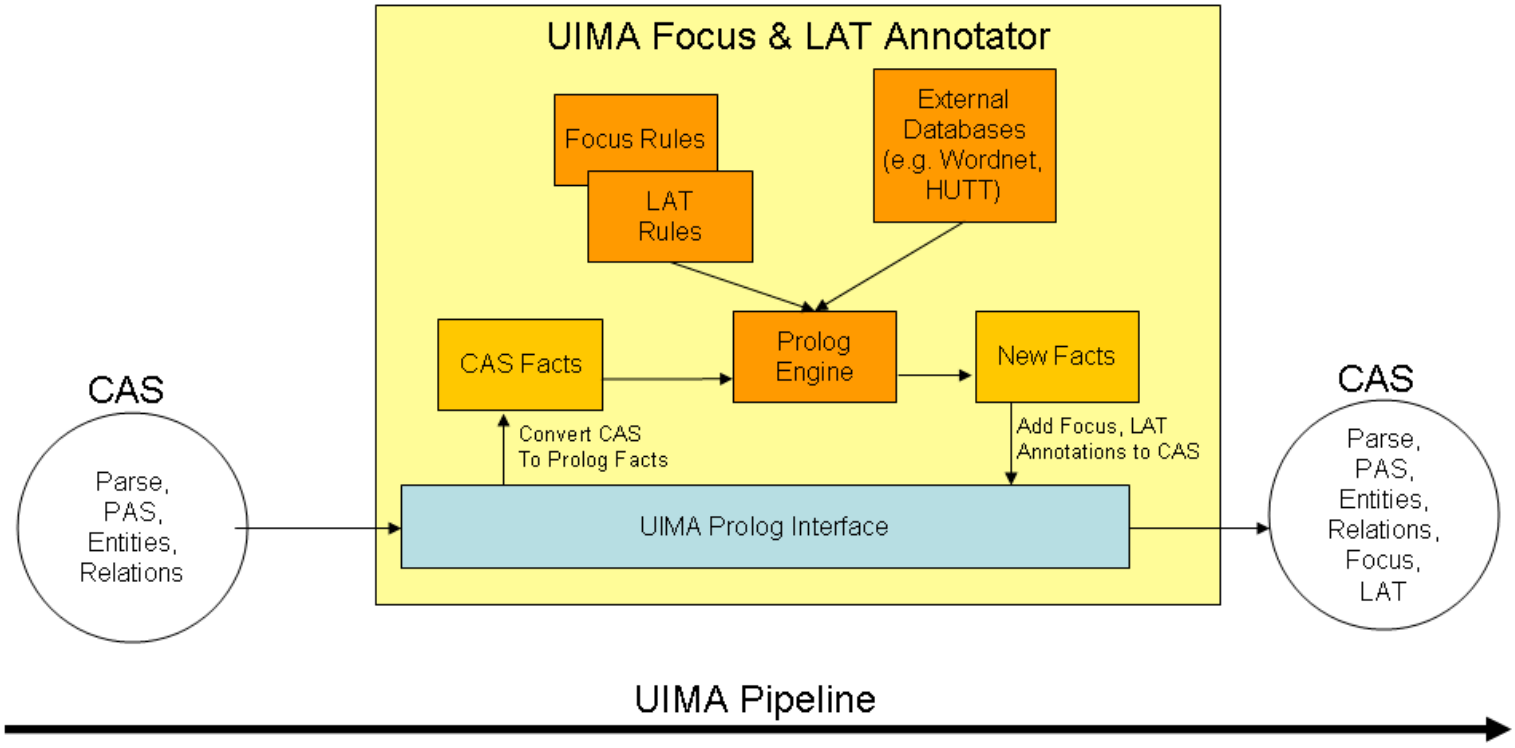
LAT

FACTOID

Question Analysis

- Rule Based and statistical approaches over ESG Annotation
- Straightforward translation of CAS to Prolog facts
- *E.g. POETS & POETRY: He was a bank clerk in the Yukon before he published "Songs of a Sourdough" in 1907*

6000 prolog clauses



LAT detection rules

- A noun phrase with determiner “this” or “these”
  - THEATRE: A new play based on *this Sir Arthur Conan Doyle canine* ***classic*** opened on the London stage in 2007.
- “this” or “these” as a pronoun
  - ’88: In April 1988 Northwest became the first U.S. air carrier to ban ***this*** on all domestic flights
- One of the pronouns “he/she/his/her/him/hers”
  - OUT WEST: ***She*** joined Buffalo Bill Cody's Wild West Show after meeting him at the Cotton Expo in New Orleans
- One of the pronouns “it/they/them/its/their”
  - ME "FIRST"!: ***It*** forbids Congress from interfering with a citizen's freedom of religion, speech, assembly or petition
- The pronoun “one”
  - 12-LETTER WORDS (200): Leavenworth, established in 1895, is a federal ***one***
- When none of the above applies, the question may have no focus, as in:
  - MOVIE TITLE PAIRS: 1999: Jodie Foster & Chow Yun-Fat

Statistical approaches are used for LAT detection in combination with rules.

Question Classes

Looked at a sample of 500 questions and refined over time

Factoid is the default class

Some QC have different ML model

Some QC have different Candidate Generation

Some QC have different pipelines

QClass	Description	Example Questions (correct answer in parentheses)	Freq.
DEFINITION	A question that contains a definition of the answer	CONSTRUCTION: It can be the slope of a roof, or the gunk used to waterproof it. (pitch) CONSTRUCTION: The name of this large beam that supports the joists literally means "something that encircles". (a girder)	14.2%
CATEGORY-RELATION	The answer has a semantic relation to the question, where the relation is specified in the category.	FORMER STATE GOVERNORS: Nelson A. Rockefeller. (New York) COUNTRIES BY NEWSPAPER: Haaretz, Yedioth Ahronoth. (Israel)	7.2%
FITB	Fill-in-the-blank – question asks for completion of a phrase	COMPLETE IT: Attributed to Lincoln: "The ___ is stronger than the bullet". (ballot) SHAKESPEARE IN LOVE: "Not that I loved Caesar less", says Brutus, "but that I loved" this city "more" (Rome)	3.8%
ABBREVI-ATION	The answer is an expansion of an abbreviation in the question	MILITARY MATTERS: Abbreviated SAS, this elite British military unit is similar to the USA's Delta Force. (the Special Air Service)	2.9%
PUZZLE	A puzzle question - the answer requires derivation, synthesis, inference, etc.	BEFORE & AFTER: 13th Century Venetian traveler who's a Ralph Lauren short sleeve top with a collar. (Marco Polo shirt) THE HIGHEST-SCORING SCRABBLE WORD: Zoom, quiz or heaven. (quiz)	2.3%
ETYMOLO-GY	A question asking for an English word derived from a foreign word having a given meaning	ARE YOU A FOOD"E"?: From the Spanish for "to bake in pastry", it's South America's equivalent of a calzone. (an empanada)	1.9%
VERB	Question asks for a verb	THE NOT-SO-DEADLY SINS: To capitalize all text in an email is an abomination that signifies the person is doing this. (shouting)	1.5%
TRANS-LATION	A question asking for translation of a word or phrase from one language to another.	FRUITS IN FRENCH: Pomme. (apple)	1.1%
NUMBER	The answer is a number	YOU NEED TO CONVERT: One eighth of a circle equals this many degrees. (45)	1.0%
BOND	The question asks for what is in common between a set of entities	EDIBLE COMMON BONDS: Mung, snap, string. (bean)	0.7%
MULTIPLE-CHOICE	The question contains multiple possible answers from which to choose the correct answer.	THE SOUTHERNMOST CAPITAL CITY: Helsinki, Moscow, Bucharest. (Bucharest) OSCAR, GRAMMY OR BOTH: Mickey Rooney. (Oscar)	0.5%
DATE	A question asking for a date or year	THE TEENS: World War I ended in November of this year. (1918)	0.3%

Table 1: Question Classes

Question Analysis: Evaluation

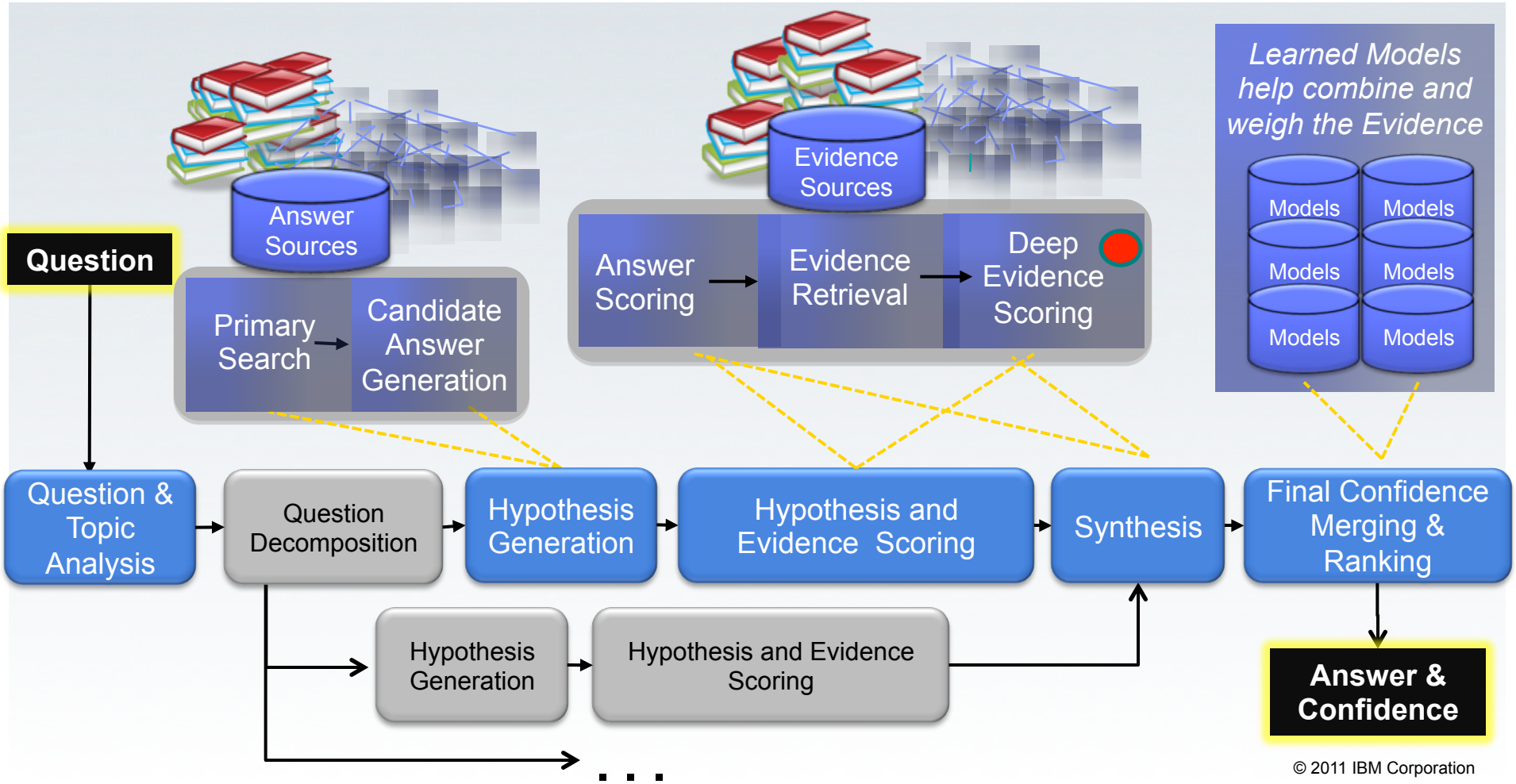
Component Level Evaluation	LAT Detection
Precision	0.829
Recall	0.766
F1	0.796
Per Question Recall	0.905

Question Classification	End To End Accuracy
No	68.1%
Yes	71.0%

Outline

- The NLP Stack
- Question Analysis
- Passage Scoring

Passage Scoring



Supporting Passage Retrieval (SPR)

Category: MICHIGAN MANIA

Clue: In 1894 C.W. Post created his warm cereal drink Postum in this Michigan city

In Deep Evidence Scoring, Watson retrieves evidence for each candidate answer, then evaluates the evidence using a large number of deep evidence scoring analytics. The evidence for a candidate answer may come from the original document or passage where the candidate answer was generated, or it may come from an evidence retrieval search performed by taking the keyword search query from Step 2, replacing the focus terms with the candidate answer, and retrieving the relevant passages that are found. The passages, or "context" in which the candidate answer occurs are evaluated as evidence to support or refute the candidate answer as the correct answer for the question.

Battle Creek

1895: In Battle Creek, Michigan, C.W. Post made the first POSTUM , a cereal beverage. Post created GRAPE-NUTS cereal in 1897, and POST TOASTIES corn flakes in 1908

The company was incorporated in 19 having developed from the earlier Postum Cereal Co. Ltd., founded by C.W. Post (1854-1914) in 1895 in Ba Creek, Mich. After a number of experiments, Post marketed his first product-the cereal beverage called Postum-in 1895

Post Foods

Post Foods, LLC, also known as Post Cereals (formerly Postum Cereals) was founded by C.W. Post. It began in 1895 with the first Postum, a "cereal beverage", developed by Post in Battle Creek, Michigan. The first cereal, Grape-Nuts, was developed in 1897

r of

became General Foods. The cereal company unit was later sold off and is now Post Foods

General Foods

1854 C. W. Post (Charles William) was born in 18 in 19 beve

General Foods' products go from breakfast (Post's cereals) to warm nightcaps (Postum, Sanka), also wash the pots and pans that its foods are cooked in (S.O.S. Scouring Pads

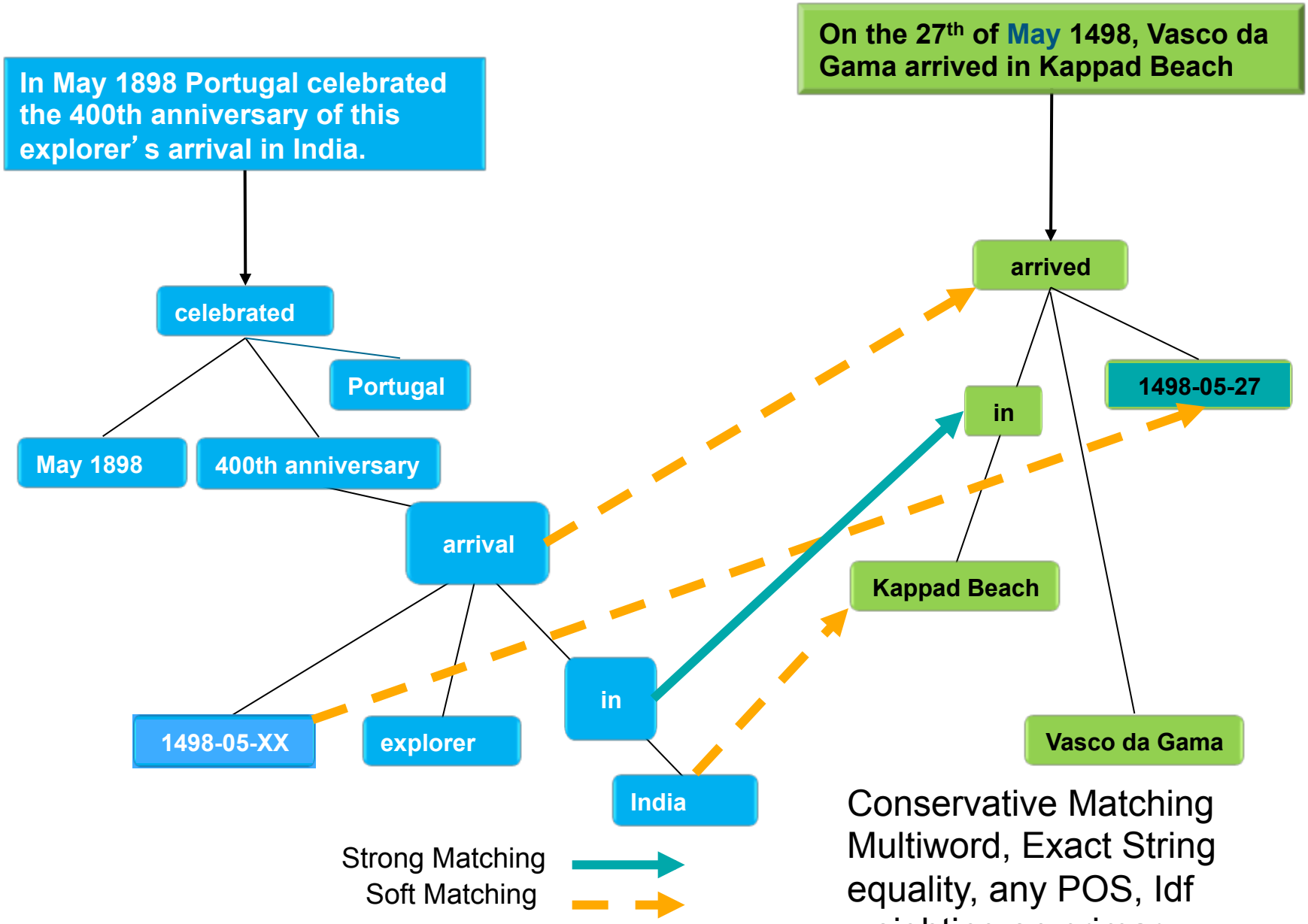
Passage scoring as a textual entailment problem

- In May 1898 Portugal celebrated the 400th anniversary of this explorer's arrival in India.
- In May 1898 Portugal celebrated the 400th anniversary of Vasco da Gama's arrival in India <- On the 27th of May 1498, Vasco da Gama landed in Kappad Beach
- In May 1898 Portugal celebrated the 400th anniversary of Gary's arrival in India <//- In May, Gary arrived in India after he celebrated his anniversary in Portugal.
- Textual Entailment is an open research issue
  - PASCAL Recognizing Textual Entailment Challenge (RTE-5) at TAC 2009
- State of the art approaches are still based on combining different similarity metrics
  - Kernel Methods
  - Edit distance
  - LSA similarity
  - Graph matching

Passage Scoring

- Apply multiple strategies to recognize textual entailment on SPR
- Passage Scoring Features
  - Passage Term Match
  - Textual Alignment
  - Skip Bigram
  - LFACS
  - LSA
  - String Kernel
- Feature are generated for each for each answer aggregating scores provided by passage scoring analytics
  - Average
  - Sum
  - Max
- Computationally expensive
  - 100 candidates per question = 2000 passages per question

Passage Term Match



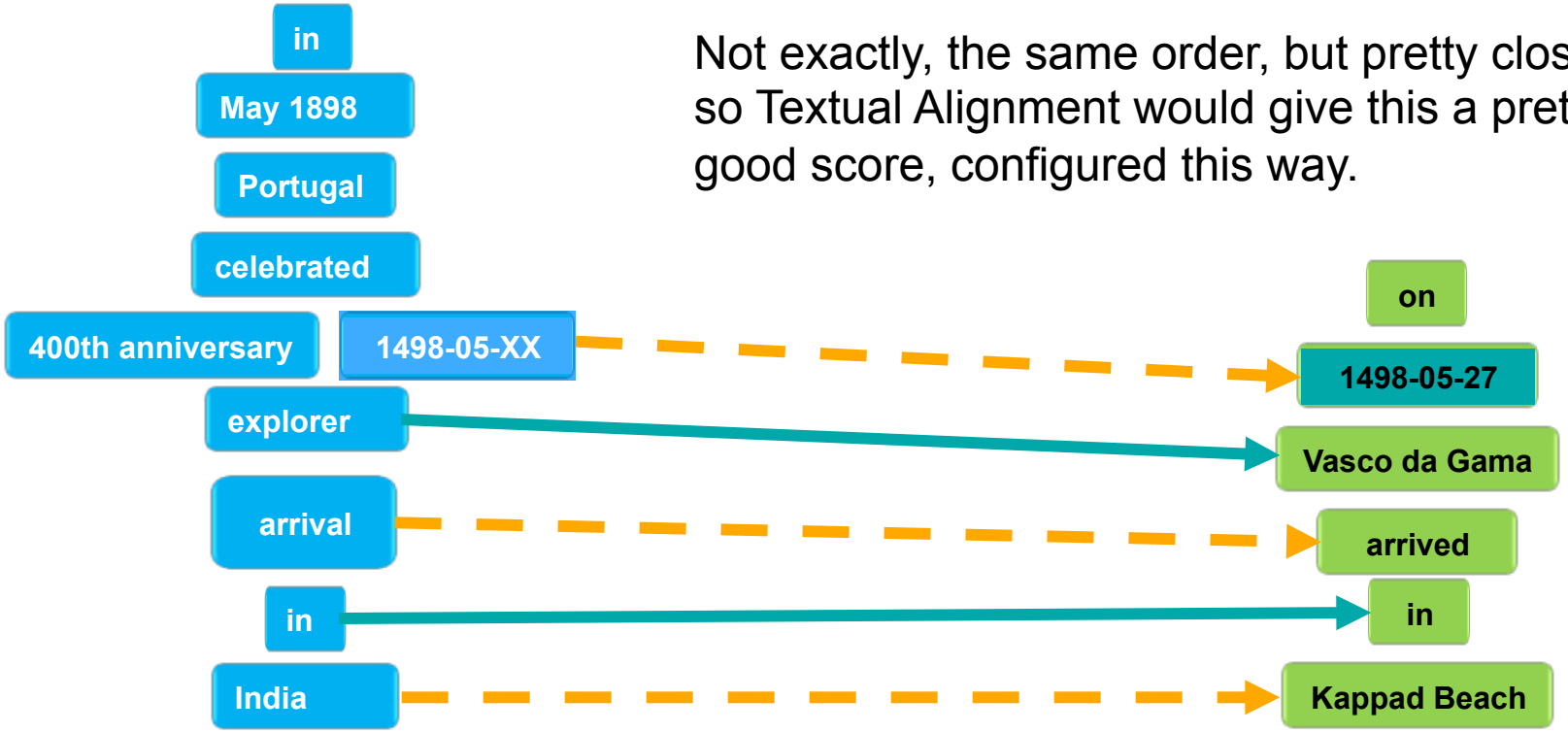
Conservative Matching
Multiword, Exact String equality, any POS, Idf weighting on primary search corpus

Textual Alignment

In May 1898 Portugal celebrated the 400th anniversary of this explorer's arrival in India.

On the 27th of May 1498, Vasco da Gama arrived in Kappad Beach

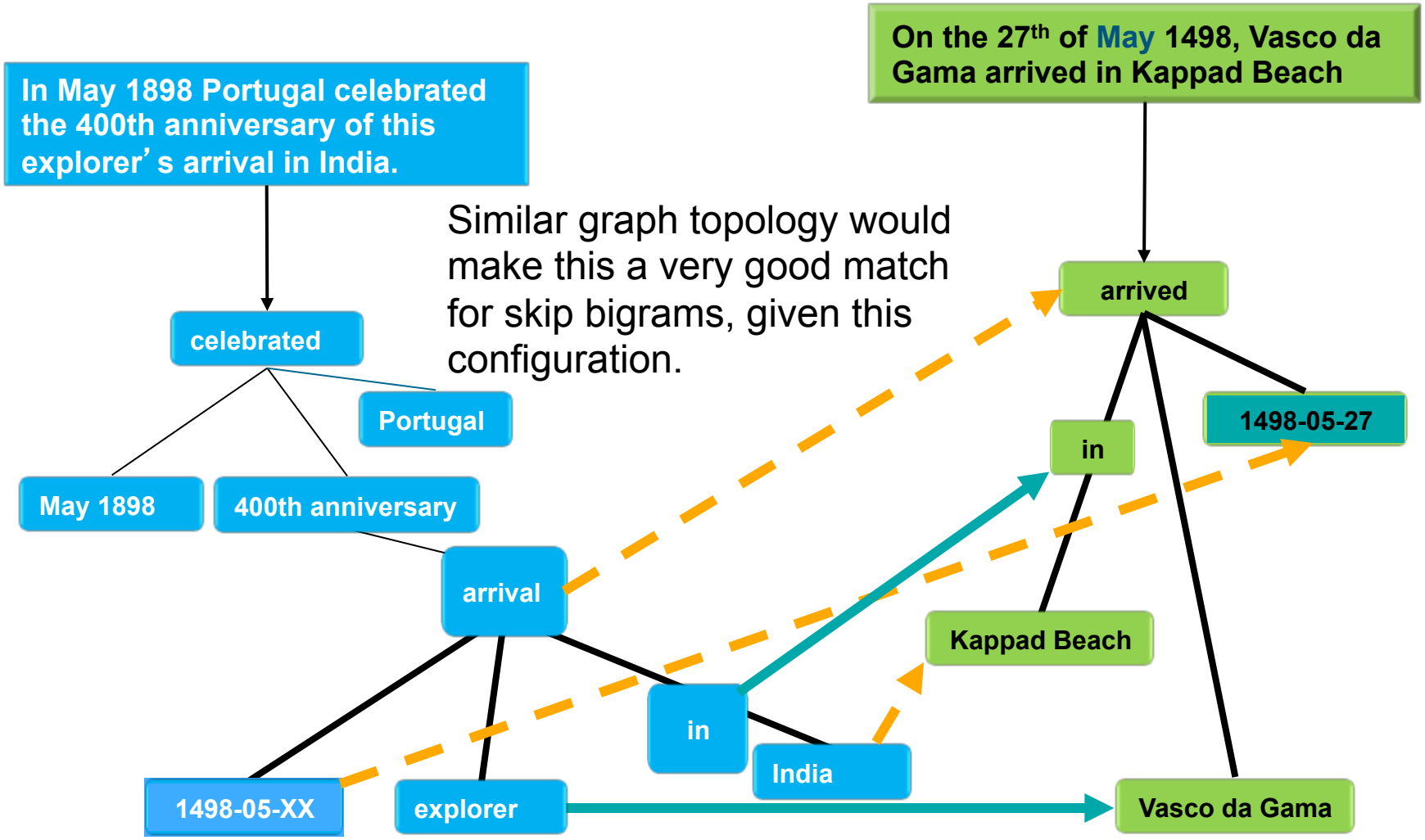
Not exactly, the same order, but pretty close, so Textual Alignment would give this a pretty good score, configured this way.



Strong Matching →
Soft Matching →

Order matters
Algorithm originally developed for gene sequences
Gap weighted score

Skip Bigram



Strong Matching →

Soft Matching →

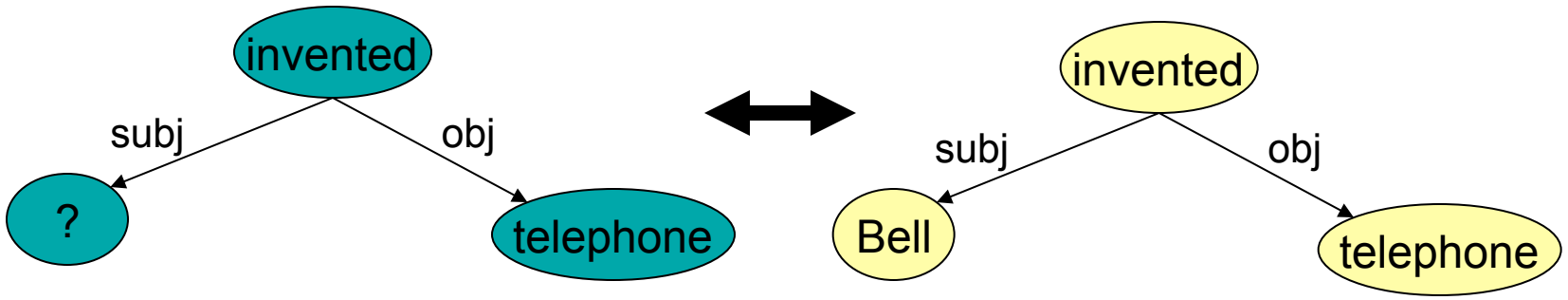
Comparison with textual alignment

Dependency graph vs sequences

Gaps of length 1 vs Gap weighted

Logical Form Answer Candidate Scorer (LFACS)

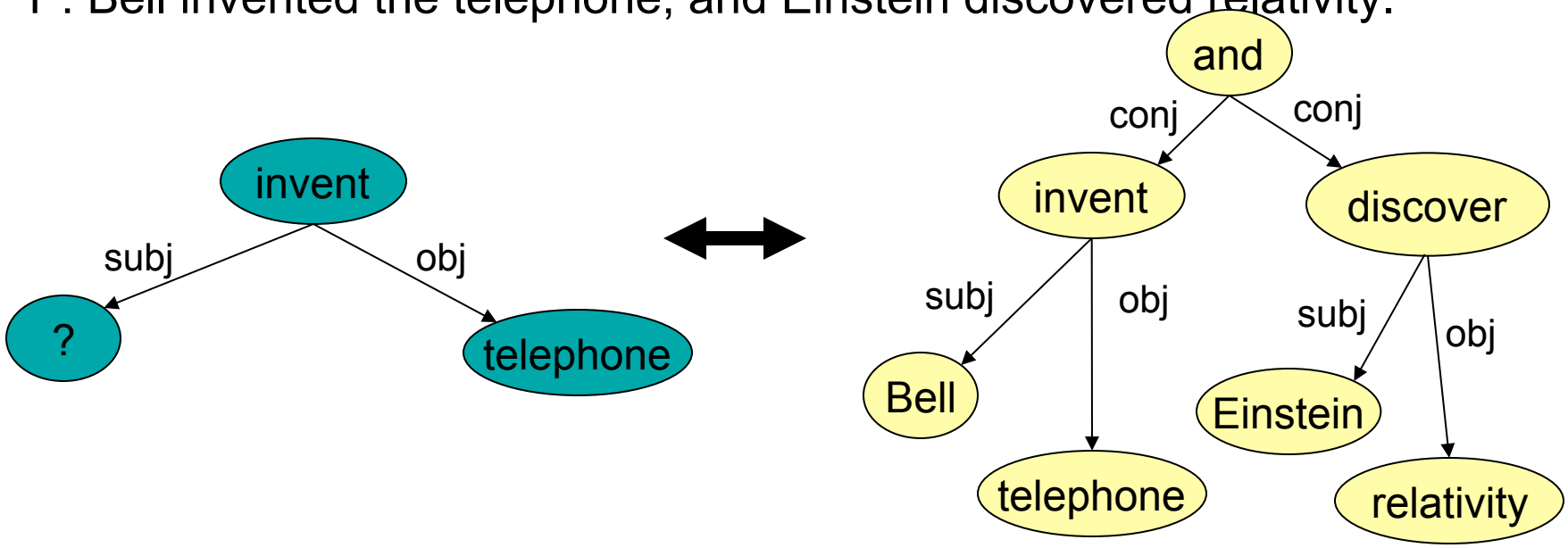
- LFACS tries to align a graph of the question to a graph of the passage:



- For complex domains (e.g., J!), there is virtually never a complete/ perfect match.
- LFACS awards partial credit based on the extent to which it is able to align portions of the graph
- LFACS is part of a suite of four passage scoring algorithms (along with Passage Term Match, Textual Alignment, and Skip Bigram)

LFACS: Focus Centered Subgraph Matching

- LFACS aligns the focus to a specified candidate answer:
- Q: Who invented the telephone?
- P: Bell invented the telephone, and Einstein discovered relativity.

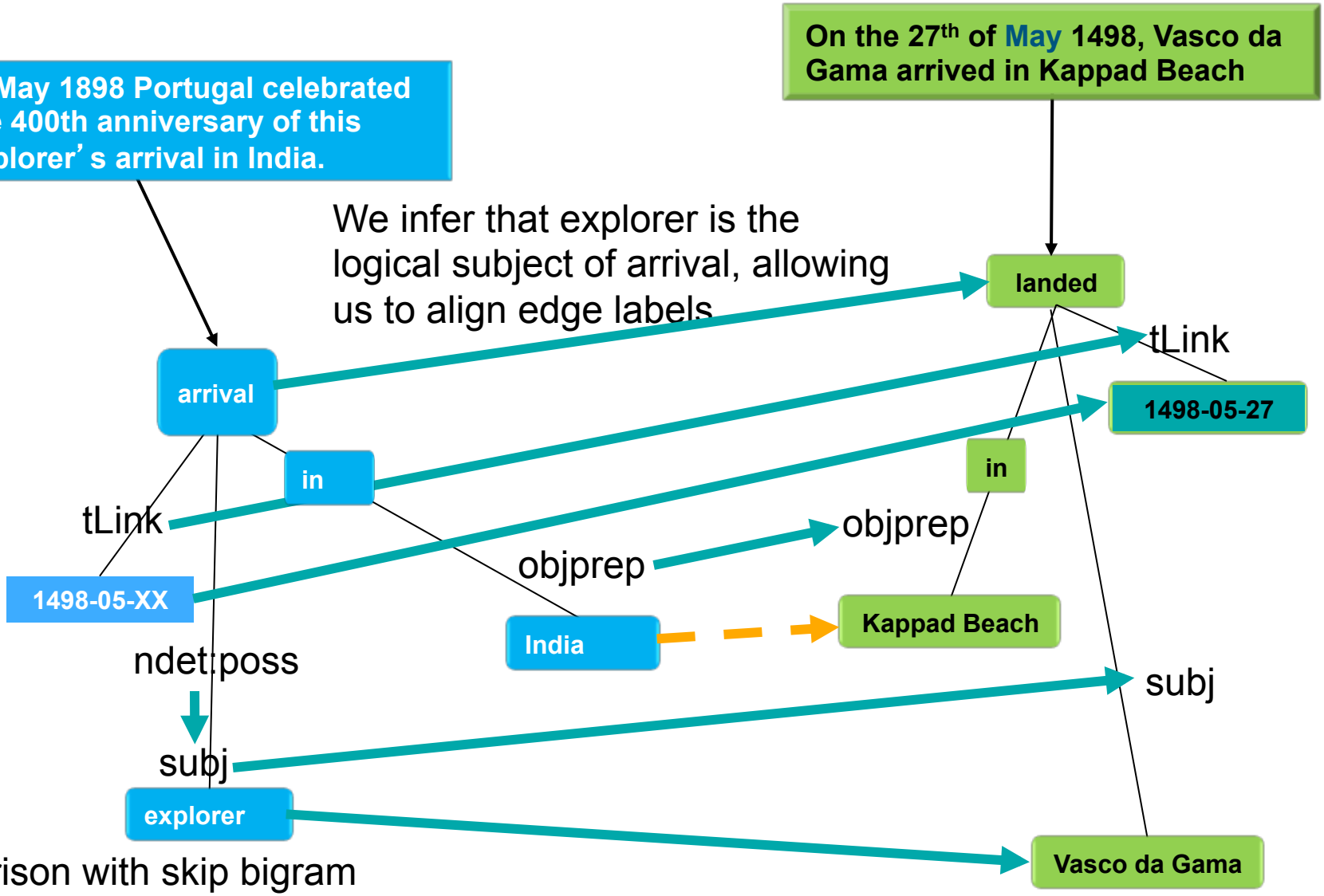


- Given this pair and the candidate answer “Bell”, LFACS will give a high score (Bell is the subj of “invent” which has obj “telephone”).
- Given this pair and the candidate answer “Einstein”, LFACS will give a score of 0 because “Einstein” is not the subject of “invent”

How is the LFACS score computed

- Attempts to match the question graph to the passage graph with the restriction that the focus of the question align with the candidate the answer.
- Node matches are performed using a complex term matcher that can be configured with various matching resources, e.g., WordNet, Wikipedia redirects.
- Edge matches are performed using a simple edge matcher that has some sense of relations and subrelations.
- **LFACS score is the sum of the IDF values of the question nodes that matched some passage nodes, weighted by the degree of match**
- "Weighted by the degree of match" is a little complicated, because there are degree of match scores for edges and nodes and some nodes that match well are only connected via nodes that match poorly.

LFACS Example



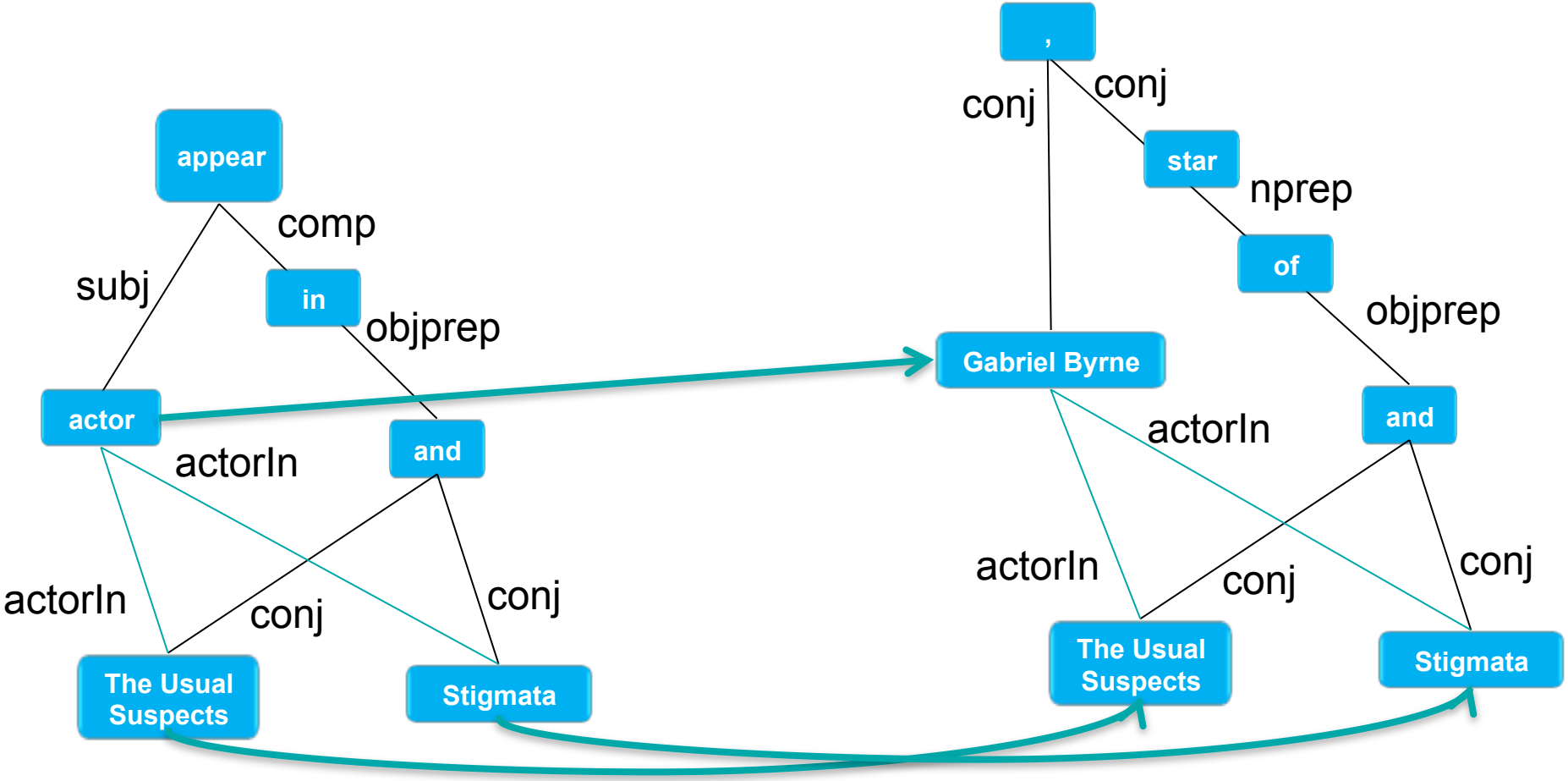
Comparison with skip bigram
Edges and nodes
No gaps
Focus is required

Strong Matching
Soft Matching

Semantic Relations in Logical Form Graphs

What actor appeared in “The Usual Suspects” and “Stigmata”?

Gabriel Byrne, star of "The Usual Suspects" and "Stigmata", …



Focus Centered Subgraph Matching is Precise but Brittle

- LFACS aligns the focus to a specified candidate answer:
- Q: Who invented the telephone?
- P: In later years, **Bell** described the **invention** of the **telephone** and linked it to his "dreaming place".
  - The passage doesn't say that Bell invented the telephone.
  - However, it is not a coincidence that the passage is talking about Bell, invention, and telephone.
  - It doesn't *prove* that Bell is the right answer, but it should be treated as *evidence* in favor of Bell being the right answer.
  - LFACS gives this passage a score of 0
- P: **Bell** is a famous **inventor**, best known for the **telephone**.
  - This passage does strongly imply that Bell invented the telephone.
  - However, "Bell" is still not the subject of the verb "invent" here. In fact there is no verb "invent"
  - LFACS gives this passage a score of 0
- P: **Bell invented** many devices including the **telephone**.
  - This passage states that Bell invented the telephone.
  - "Bell" is the subject of the verb "invent," but the object of "invent" is "devices"
  - LFACS gives this passage partial credit (for "invent" but not "telephone")

Kernel Methods for passage scoring

Kernels are similarity functions that can be applied to measure the similarity between two text

- Linear Kernel (BOW)
- Sequences (String Kernel, Word sequence kernel)
- Syntactic Structures (Tree Kernel)
- Similarity in a topic model (Domain Kernel, LSI)

A kernel is a function $K : \mathcal{X} \times \mathcal{X} -> \mathbb{R}$ such that

(18) $$K(x_i, x_j) = (def)\langle \mathbf{\Phi}(x_i), \mathbf{\Phi}(x_j)\rangle$$

where $\mathbf{\Phi} : \mathcal{X} -> \mathcal{K}$ is a feature mapping.

Kernel trick: equivalent (and more efficient) formulation in the instance space, avoid explicit feature mapping

String kernel Fixed Length Subsequence Kernel

it counts the number of common subsequences of length p

(26) $$\Phi_u^p(s) = |\{\mathbf{i} : u = s(\mathbf{i})\}|, u \in \Sigma^p$$

(27) $$k_p(s,t) = \langle \Phi^p(s), \Phi^p(t) \rangle = \sum_{u \in \Sigma^p} \Phi_u^p(s) \Phi_u^p(t)$$

	c-a	c-r	a-r	c-t	a-t	c-u	u-t
Φ^2(car)	1	1	1	0	0	0	0
Φ^2(cat)	1	0	0	1	1	0	0
Φ^2(cut)	0	0	0	1	0	1	1

Gap Weighted Subsequence Kernel

It assigns different weights to sequences, according to how spread they are in the original strings

(28) $$\Phi_u^p(s) = \sum_{\mathbf{i}:u=s(\mathbf{i})} \lambda^{l(\mathbf{i})}, u \in \Sigma^p$$

$\lambda \in [0, 1]$: When $\lambda = 1$ this kernel is equivalent to the fixed length subsequence kernel, if $\lambda -> 0$ it approximates the p-spectrum kernel

	c-a	c-r	a-r	c-t	a-t	c-u	u-t
ϕ(car)	λ^2	λ^3	λ^2	0	0	0	0
ϕ(cat)	λ^2	0	0	λ^3	λ^2	0	0
ϕ(cut)	0	0	0	λ^3	0	λ^2	λ^2

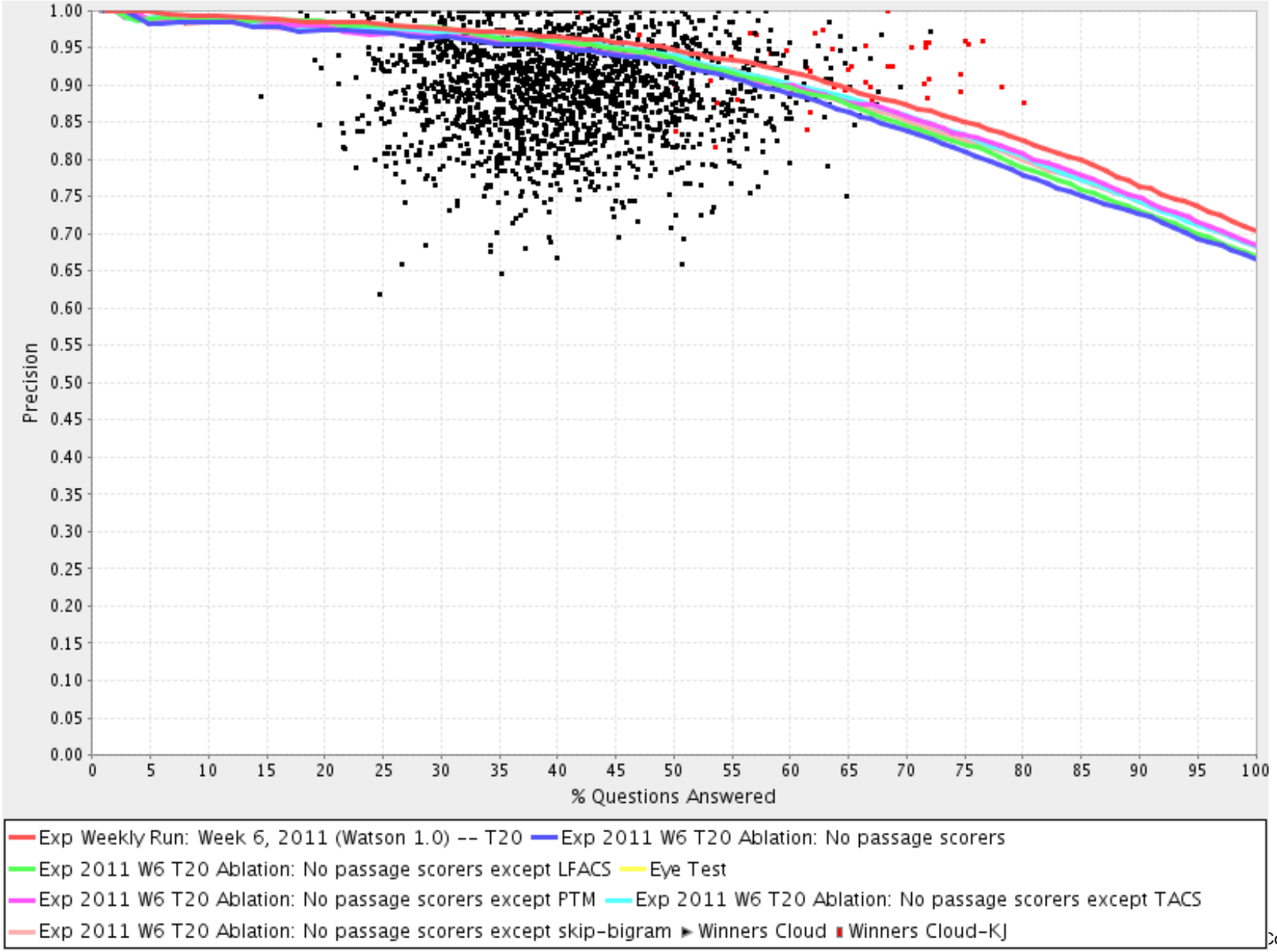
String Kernel Passage Scorer

- Measure the similarity between question and supporting passage where both focus and the candidate answer are replaced with a placeholder (FOCUS)
- Word Sequence Kernel (words are used instead of letters)
- Using ngrams of length 2 and 3
- Optimization lambda pruning (do not consider subsequences of span > k)
- Lemmatized forms
  - Q Who invented the telephone?
  - P1 **Bell invented** many devices including the **telephone**
- sim (FOCUS invent the telephone, FOCUS invent many device include the telephone)
  - Match (FOCUS invent _ telephone, FOCUS invent _ _ _ _ telephone)

LSA Passage Scorer

- Will be presented in Distributional Semantics Lesson

Passage Scoring evaluation



Outline

- The NLP Stack
- Question Analysis
- Passage Scoring