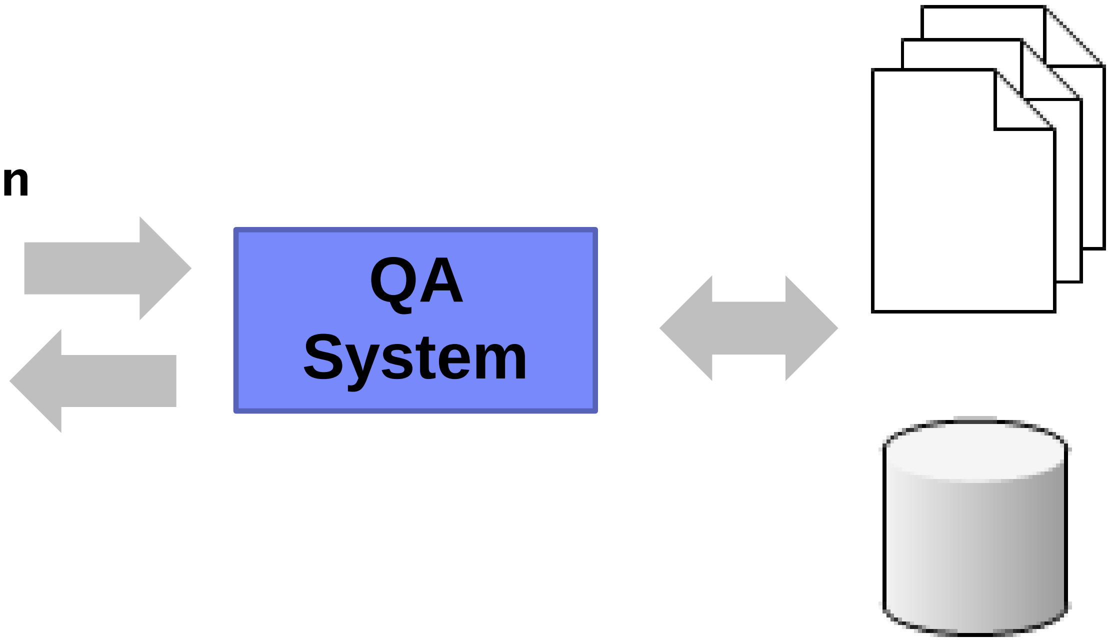
Question Answering (II)
the State of the Arts

Instructor: Huan Sun
Computer Science
University of California at Santa Barbara

Recap: What is Question Answering?

Text Corpora

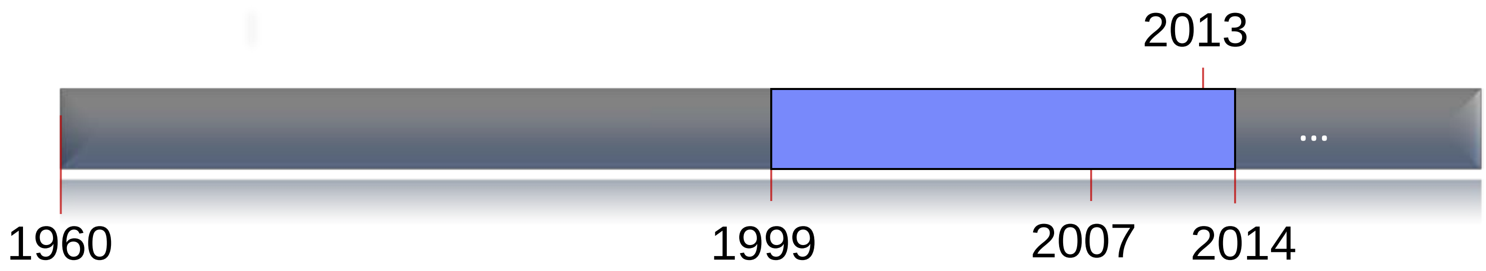
Question: Who is the daughter of Bill Clinton married to?



Answer: Marc Mezvinsky

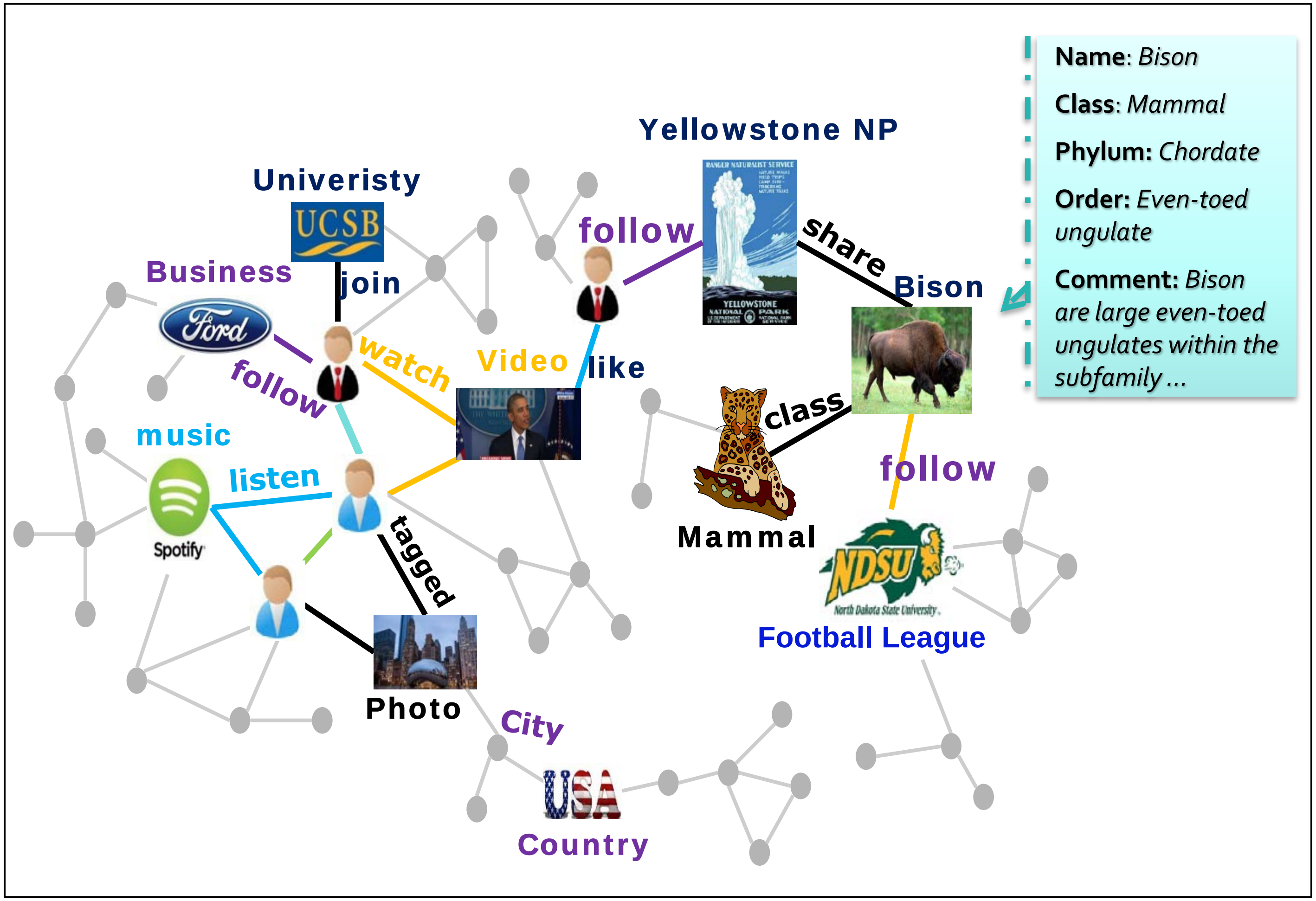
Database

CS290D covers



- Open domain QA (TREC, 1999~2007)
- QA over linked data (~2007-)
- Recent Developments (~2013-)
  - April 29th

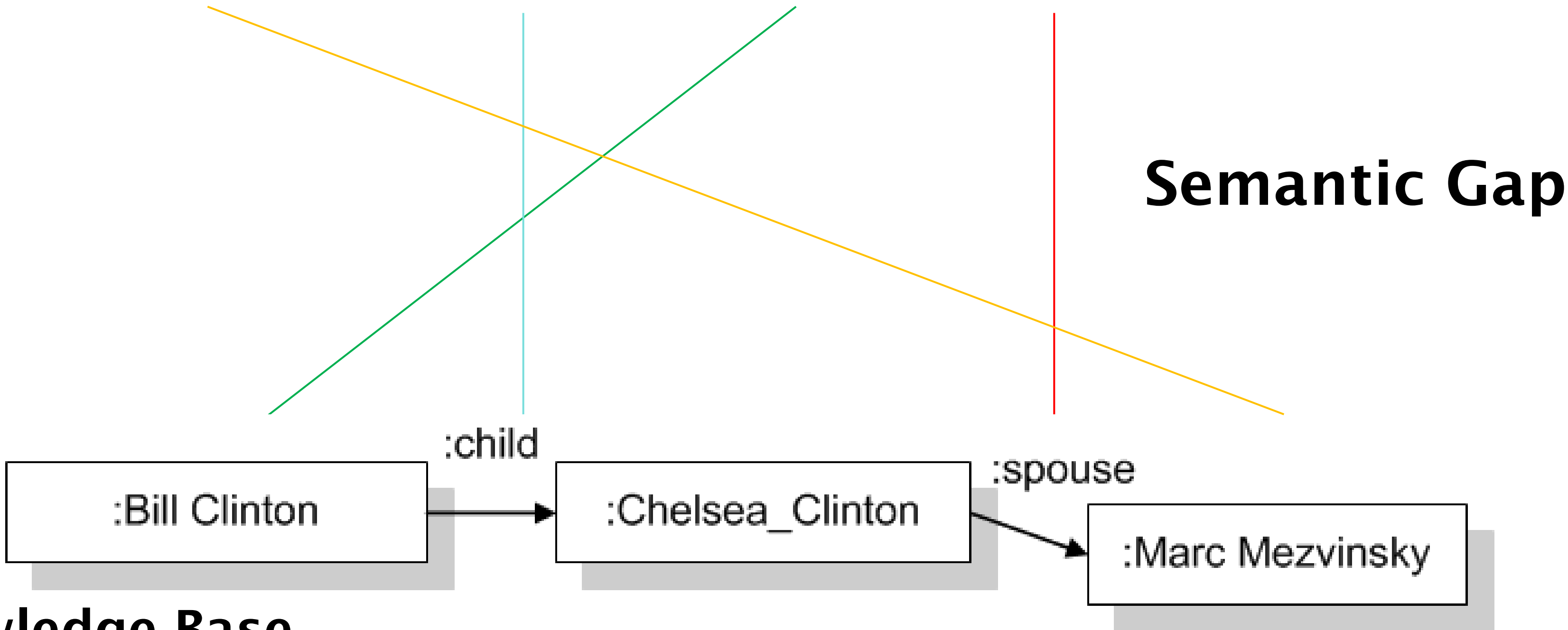
Blossom of Large-scale Knowledge Bases



Courtesy of Shengqi Yang, UCSB

Representation Mismatch

Question: Who is the daughter of Bill Clinton married to ?



Knowledge Base

How to bridge the gap?

Recent Methodologies for QA

- Semantic Parsing
  - Percy Liang, Stanford
- Embedding-based
  - Jason Weston, Facebook
- Deep Neural Networks
  - Hal Daume III, UMD
- Graph Querying
  - Lei Zou, Peking University
  - Haixun Wang, Google
  - Our group

Semantic Parsing via Paraphrasing

Berant et al., ACL 2014

Slides in this section were largely adapted from
http://www-nlp.stanford.edu/joberant/homepage_files/talks/facebook_jun14.pdf
by Jonathan Berant and Percy Liang

Semantic Parsing

- Definition
  - Mapping natural language utterances into logical forms
  - Logical forms, such as lambda calculus, lambda DCS (Dependency-based Compositional Semantics)

  e.g.,

  Utterance: *"people who have lived in Seattle"*

  Logical form (lambda calculus): $\lambda x.\exists e.\texttt{PlacesLived}(x, e) \wedge \texttt{Location}(e, \texttt{Seattle})$

  Logical form (lambda DCS): $\texttt{PlacesLived.Location.Seattle}$

- Lambda DCS
  - To build a natural language interface to Freebase

Lambda DCS Preliminaries

- Knowledge base $\mathcal{K}$
  - $\mathcal{E}$: the set of entities

    $\texttt{Seattle}$
  - $\mathcal{P}$: the set of predicates

    $\texttt{PlaceOfBirth}$
  - $\mathcal{K}$: the set of assertions, $\mathcal{K} \subset \mathcal{E} \times \mathcal{P} \times \mathcal{E}$

    $(\texttt{BillGates}, \texttt{PlaceOfBirth}, \texttt{Seattle})$

- Lambda DCS form z, (logical form)
  - $[\![z]\!]_{\mathcal{K}}$: denotations of z.

Lambda DCS Preliminaries

- Basic lambda DCS logical forms
  - Unary base case: an entity

$$\texttt{Seattle}$$

  - Binary base case: a predicate

$$\texttt{PlaceOfBirth}$$

  - Join, “people who were born in Seattle”

$$\texttt{PlaceOfBirth.Seattle}$$

  - Intersection

$$\texttt{Profession.Scientist} \sqcap \texttt{PlaceOfBirth.Seattle}$$

  - Aggregation (e.g., count, min, max)

$$\texttt{count}(\texttt{Type.USState}).$$

Semantic Parsing for Question Answering

Who did Humphrey Bogart marry in 1928?

semantic parsing

$\text{Type.Person} \sqcap \text{Marriage.(Spouse.HumphreyBogart} \sqcap \text{StartDate.1928)}$

execute logical form

Mary Philips

Traditional Statistical Semantic Parsing

Supervision: manually annotated logical forms

What's California's capital?	Capital.California
How long is the Mississippi river?	RiverLength.Mississippi
...	...

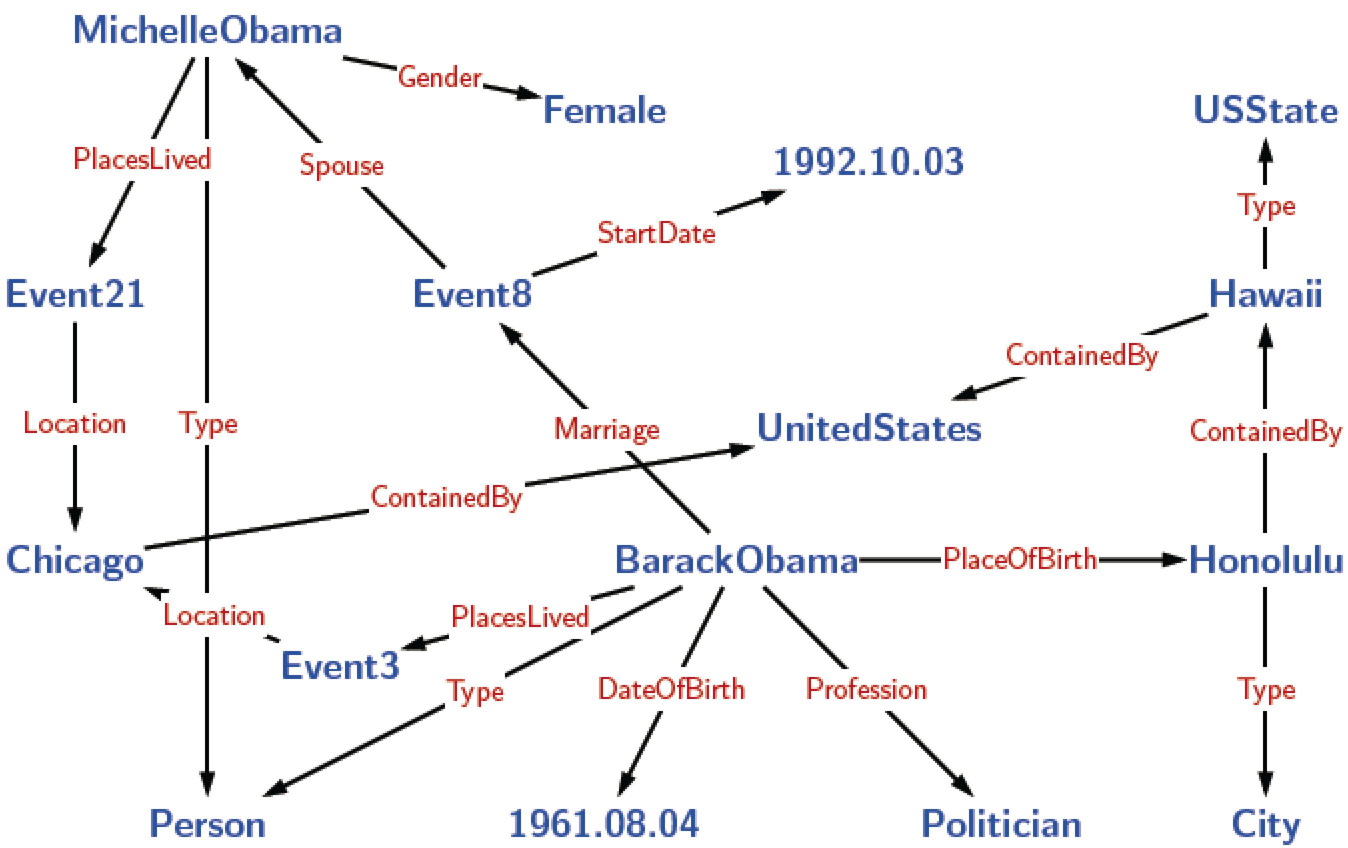
Limitations:

- Requires experts — slow, expensive, does not scale!
- Restricted to limited domains

[Zelle and Mooney, 1996; Zettlemoyer and Collins, 2005; ...]

Scaling to Large Knowledge Bases

Freebase



41M entities (nodes)

19K properties (edge labels)

596M assertions (edges)

Overview and Notation

What languages do people in Brazil use?

x: input question

Overview and Notation

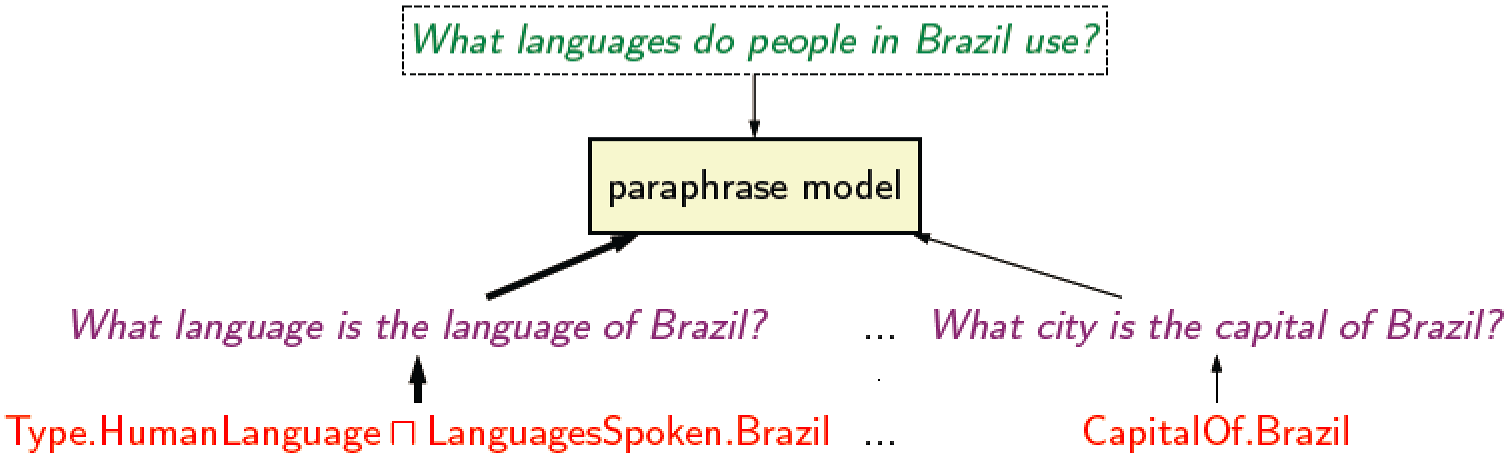
What languages do people in Brazil use?

Type.HumanLanguage $\sqcap$ LanguagesSpoken.Brazil ... CapitalOf.Brazil

x: input question

Z_x: candidate logical forms (candidate logical form generation)

Overview and Notation

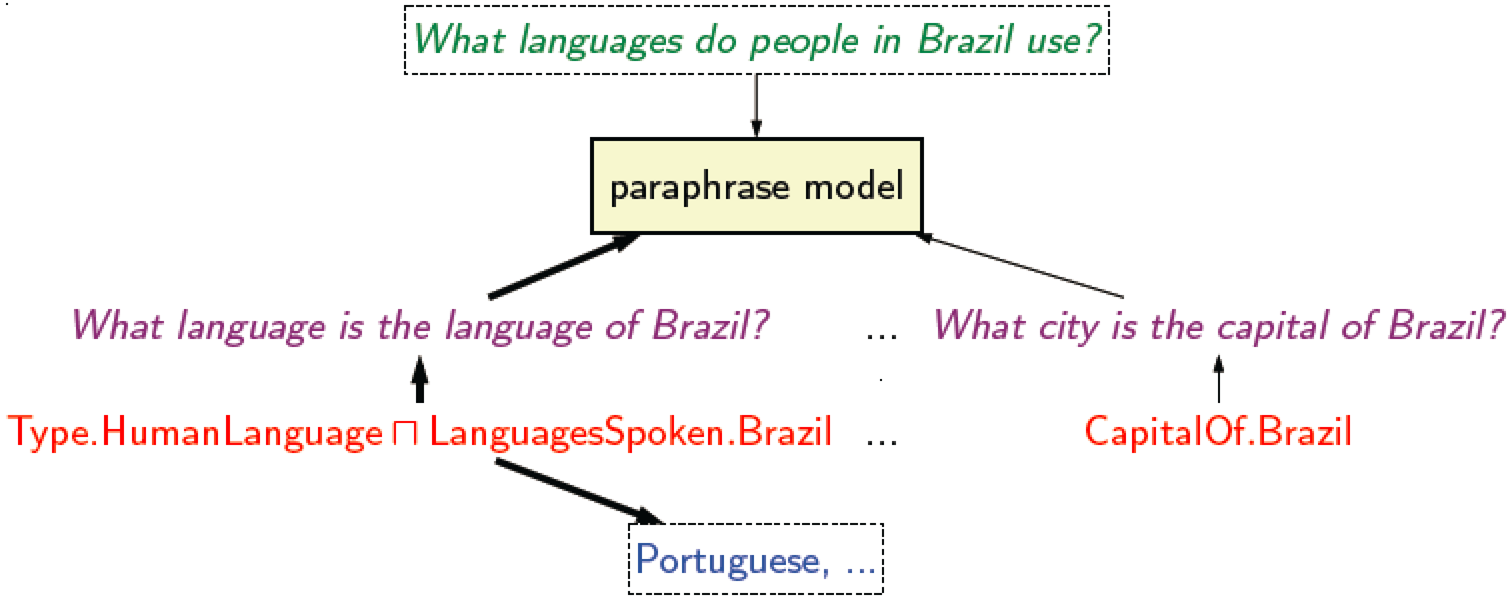


x: input question

Z_x: candidate logical forms (candidate logical form generation)

C_z: generated canonical utterances (canonical utterance generation)

Overview and Notation



x: input question

Z_x: candidate logical forms (candidate logical form generation)

C_z: generated canonical utterances (canonical utterance generation)

y: answer

Model

- Given a pair of a candidate logical form z and a canonical utterance c

Model: distribution over logical forms and canonical utterances

$$p_\theta(c, z \mid x) = \frac{\exp(\phi(x,c,z)^\top \theta)}{\sum_{z' \in Z_x, c' \in C_z} \exp(\phi(x,z',c')^\top \theta)}$$

Decomposition to paraphrase model and logical form model:

$$\phi(x, c, z)^\top \theta = \phi_{\text{pr}}(x, c)^\top \theta_{\text{pr}} + \phi_{\text{lf}}(x, z)^\top \theta_{\text{lf}}.$$

Need to estimate parameteres θ_{pr} and θ_{lf}

Learning

Training data: $\{(x_i, y_i)\}_{i=1}^n$

Learning

Training data: $\{(x_i, y_i)\}_{i=1}^n$

Objective function:

$$p_\theta(y \mid x) = \sum_{z \in Z_x : y = [z]_\mathcal{K}} \sum_{c \in C_z} p_\theta(c, z \mid x)$$

$$O(\theta) = \sum_{i=1}^{n} \log p_\theta(y_i \mid x_i) - \lambda \|\theta\|_1$$

How to Generate Candidate Logical Forms (Z_x) ?

- Growing logical forms around entities

What countries in the world speak Arabic?

ArabicAlphabet ArabicLang

- Following logical form templates

How to Generate Candidate Logical Forms (Z_x) ?

- Growing logical forms around entities

What countries in the world speak Arabic?

ArabicAlphabet   ArabicLang

- Following logical form templates

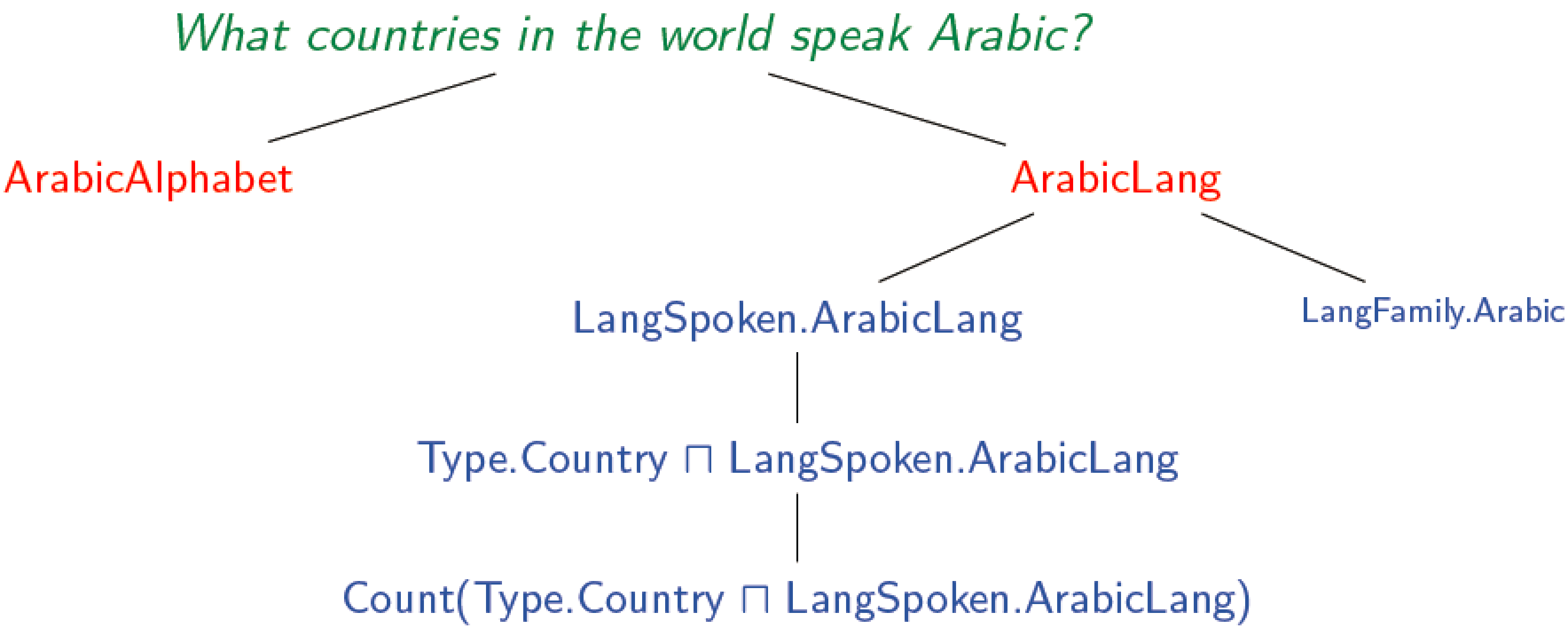
Template	Example	Question
$p.e$	Directed.TopGun	*who directed Top Gun*
$p_1.p_2.e$	Employment.EmployerOf.SteveBalmer	*Where does Steve Balmer work?*
$p.(p_1.e_1 \sqcap p_2.e_2)$	Character.(Actor.BradPitt⊓Film.Troy)	*Who did Brad Pitt play in Troy?*
Type.$t \sqcap z$	Type.Composer⊓SpeakerOf.French	*What composers spoke French?*
count(z)	count(BoatDesigner.NatHerreshoff)	*How many ships were designed by Nat Herreshoff?*

How to Generate Candidate Logical Forms (Z_x) ?

- Growing logical forms around entities
- Following logical form templates

How to Generate Candidate Logical Forms (Z_x) ?

- Growing logical forms around entities
- Following logical form templates



How to Generate Canonical Utterances (C_z) ?

- Following generation rules

	$d(p)$ Categ.	Rule	Example
$p.e$	NP	WH $d(t)$ has $d(e)$ as NP ?	*What **election contest** has **George Bush** as **winner?***
	VP	WH $d(t)$ (AUX) VP $d(e)$?	*What **radio station serves area New-York**?*
	PP	WH $d(t)$ PP $d(e)$?	*What **beer from region Argentina**?*
	NP VP	WH $d(t)$ VP the NP $d(e)$?	*What **mass transportation system served** the **area Berlin**?*
$\mathbf{R}(p).e$	NP	WH $d(t)$ is the NP of $d(e)$?	*What **location** is the **place of birth** of **Elvis Presley**?*
	VP	WH $d(t)$ AUX $d(e)$ VP ?	*What **film** is **Brazil featured in**?*
	PP	WH $d(t)$ $d(e)$ PP ?	*What **destination Spanish steps near travel destination**?*
	NP VP	WH NP is VP by $d(e)$?	*What **structure** is **designed** by Herod?*

d(t), d(e), and d(p) are respectively denoted by Freebase descriptions (natural language phrases) for the type, entity, and property.

Rules are based on

1. Entity being asked is subject or object of ***p***
2. Parse tree of predicates' descriptions

How to Generate Canonical Utterances (C_z) ?

- Growing logical forms around entities

- Following generation rules

Type.Country⊓LangSpoken.ArabicLang

country *spoken in* *languages spoken* *Arabic language*

syntactic analysis

What country is Arabic language spoken in?

What country spoken the languages Arabic language?

Model

- Given a pair of a candidate logical form z and a canonical utterance c

Model: distribution over logical forms and canonical utterances

$$p_\theta(c, z \mid x) = \frac{\exp(\phi(x,c,z)^\top \theta)}{\sum_{z' \in Z_x, c' \in C_z} \exp(\phi(x,z',c')^\top \theta)}$$

Decomposition to paraphrase model and logical form model:

$$\phi(x, c, z)^\top \theta = \underline{\phi_{\text{pr}}(x, c)}^\top \theta_{\text{pr}} + \underline{\phi_{\text{lf}}(x, z)}^\top \theta_{\text{lf}}.$$

$\phi_{\text{lf}}(x, z)$:

Features extracted based on the logical form and the input utterance

$\phi_{\text{pr}}(x, c)$: paraphrase model

Paraphrase model

Question: *What countries in the world speak Arabic?*

Canonical utterance: *What country is Arabic language spoken in?*

Simple paraphrase model utilizing **a lot of text**

- Assocation model
- Vector space model

$$\phi_{\text{pr}}(x, c)^\top \theta_{\text{pr}} = \phi_{\text{as}}(x, c)^\top \theta_{\text{as}} + \phi_{\text{vs}}(x, c)^\top \theta_{\text{vs}}.$$

Association Generation

Paralex dataset (Fader et al., 2013)

- 18M word aligned question pairs
- Generated through links in WikiAnswers

Who wrote the Winnie the Pook books?	*Who is poohs creator?*
What relieves a hangover?	*What is the best cure for a hangover?*
How do you say Santa Clause in Sweden?	*Say santa clause in sweden?*

Association Generation

Paralex dataset (Fader et al., 2013)

- 18M word aligned question pairs
- Generated through links in WikiAnswers

Who wrote the Winnie the Pook books?	*Who is poohs creator?*
What relieves a hangover?	*What is the best cure for a hangover?*
How do you say Santa Clause in Sweden?	*Say santa clause in sweden?*

Consistent phrase pair heuristic (Och and Ney, 2004):

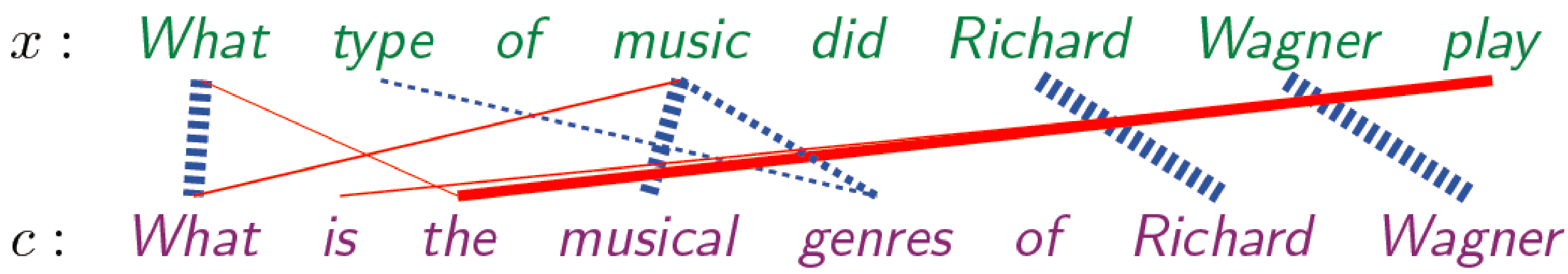
phrase association table: *type of music* ⇔ musical genre
born in ⇔ birth place

Associate words with same POS tag or that are linked through WordNet

Association model

Association: pair of spans $(x_{ij}, c_{i'j'})$

type of music ⇔ musical genre



Generate all associations and extrct features: Based on phrase association table

identical lemma	3
type of music $\wedge$ musical genre	1
play $\wedge$ the	1
WN derivation	1

…

$$\phi_{\text{pr}}(x,c)^\top\theta_{\text{pr}} = \phi_{\text{as}}(x,c)^\top\theta_{\text{as}} + \phi_{\text{vs}}(x,c)^\top\theta_{\text{vs}}.$$

Vector Space Generation

Assocations disadvantage: coverage

Train word vectors $v(w)$:

C: content words in utterance x

$$v(x) = \frac{1}{|C|} \sum_{x_i \in C} v(x_i)$$

Learn a matrix W to estimate "similarity" score

$$s(x, c) = v(x)^\top W v(c)$$

Let $\theta_{\text{vs}} = \text{vec}(W)$, $\phi_{\text{vs}}(x, c) = \text{vec}(v_x v_c^\top)$

$\phi_{\text{vs}}(x, c)^\top \theta_{\text{vs}}$ = $s(x, c)$

Review: Model

- Given a pair of a candidate logical form z and a canonical utterance c

Model: distribution over logical forms and canonical utterances

$$p_\theta(c, z \mid x) = \frac{\exp(\phi(x,c,z)^\top \theta)}{\sum_{z' \in Z_x, c' \in C_z} \exp(\phi(x,z',c')^\top \theta)}$$

Decomposition to paraphrase model and logical form model:

$$\phi(x, c, z)^\top \theta = \phi_{\text{pr}}(x, c)^\top \theta_{\text{pr}} + \phi_{\text{lf}}(x, z)^\top \theta_{\text{lf}}.$$

Where, $\phi_{\text{pr}}(x, c)^\top \theta_{\text{pr}} = \phi_{\text{as}}(x, c)^\top \theta_{\text{as}} + \phi_{\text{vs}}(x, c)^\top \theta_{\text{vs}}.$

Need to estimate parameteres θ_{pr} and θ_{lf}

Review: Learning

Training data: $\{(x_i, y_i)\}_{i=1}^{n}$

Objective function:

$$p_\theta(y \mid x) = \sum_{z \in Z_x : y = [z]_{\mathcal{K}}} \sum_{c \in C_z} p_\theta(c, z \mid x)$$

$$O(\theta) = \sum_{i=1}^{n} \log p_\theta(y_i \mid x_i) - \lambda \|\theta\|_1$$

Recap

What languages do people in Brazil use?

x: input question

Recap

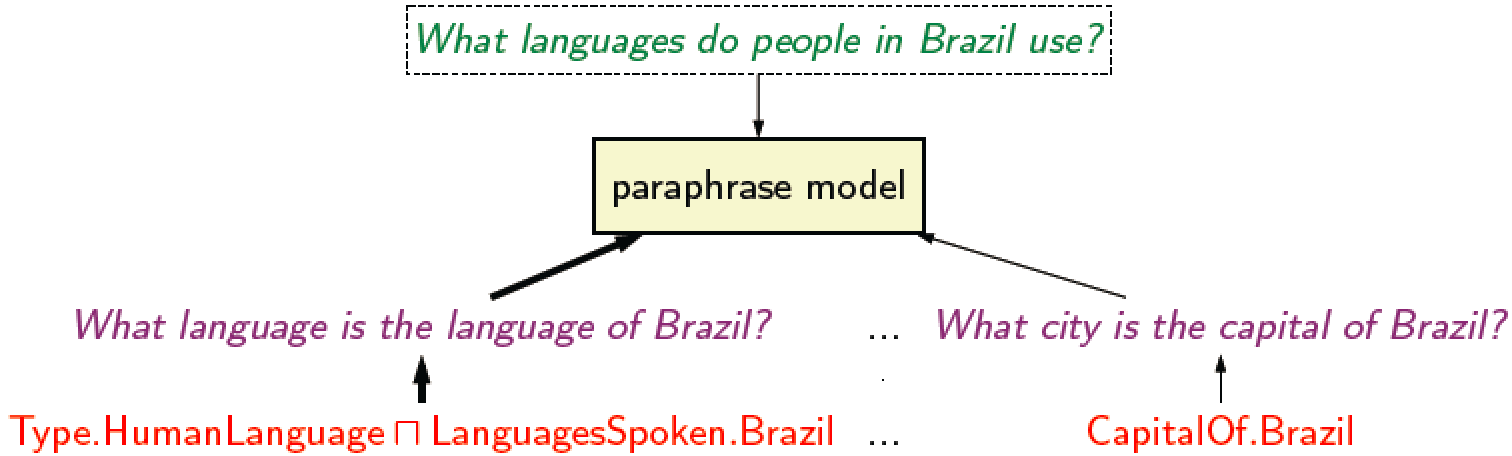
What languages do people in Brazil use?

$\text{Type.HumanLanguage} \sqcap \text{LanguagesSpoken.Brazil}$ ... CapitalOf.Brazil

x: input question

Z_x: candidate logical forms (candidate logical form generation)

Recap

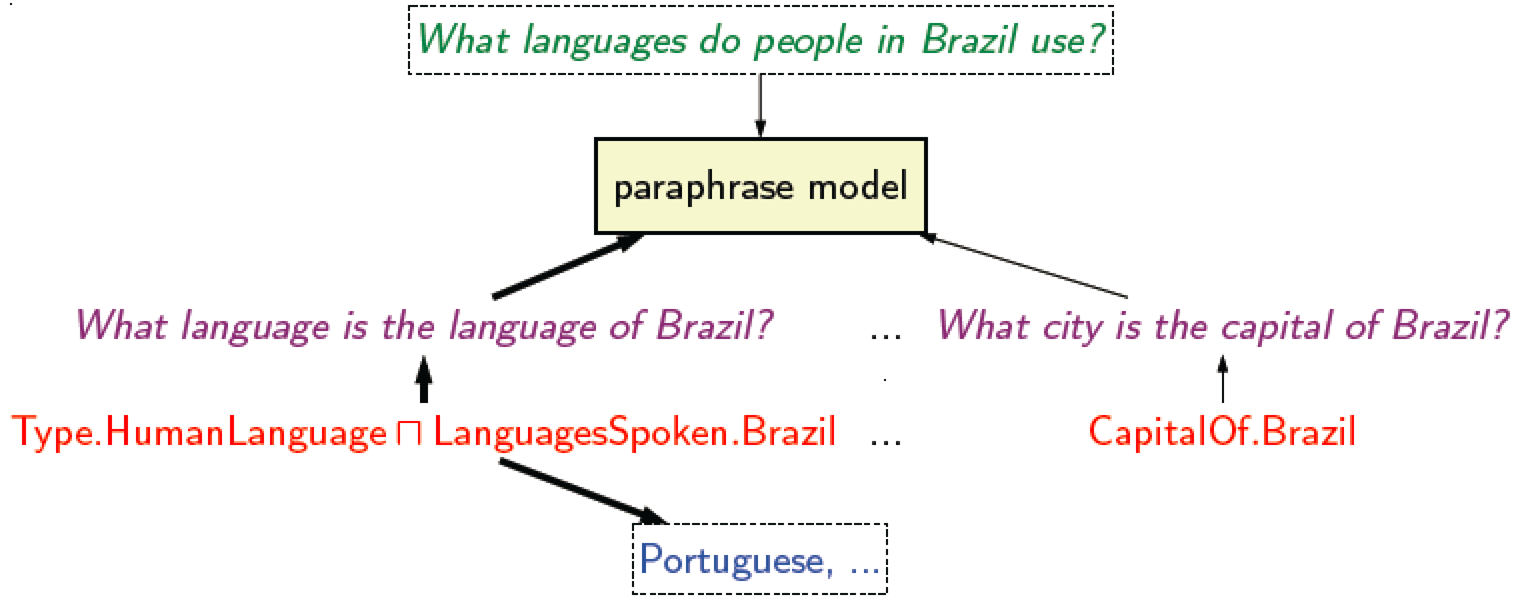


x: input question

Z_x: candidate logical forms (candidate logical form generation)

C_z: generated canonical utterances (canonical utterance generation)

Recap



x: input question

Z_x: candidate logical forms (candidate logical form generation)

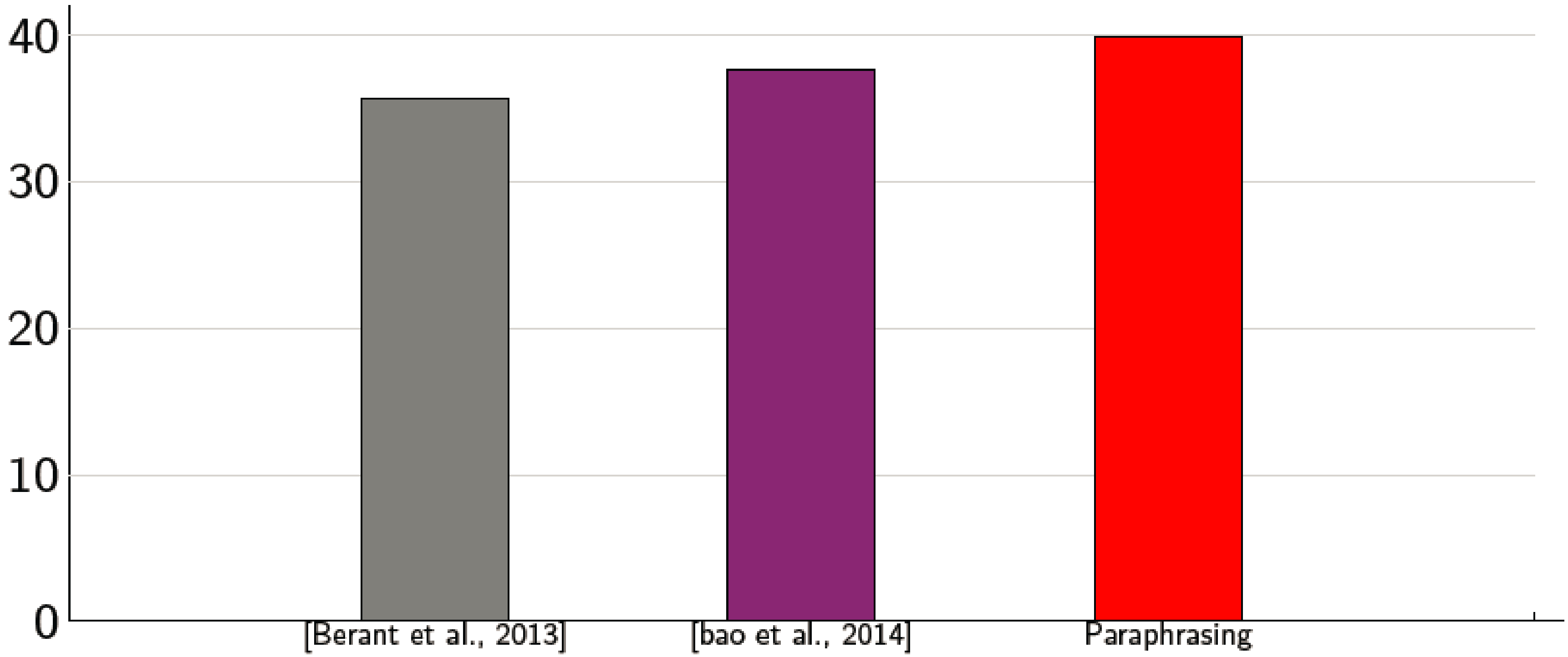
C_z: generated canonical utterances (canonical utterance generation)

y: answer

Experiments

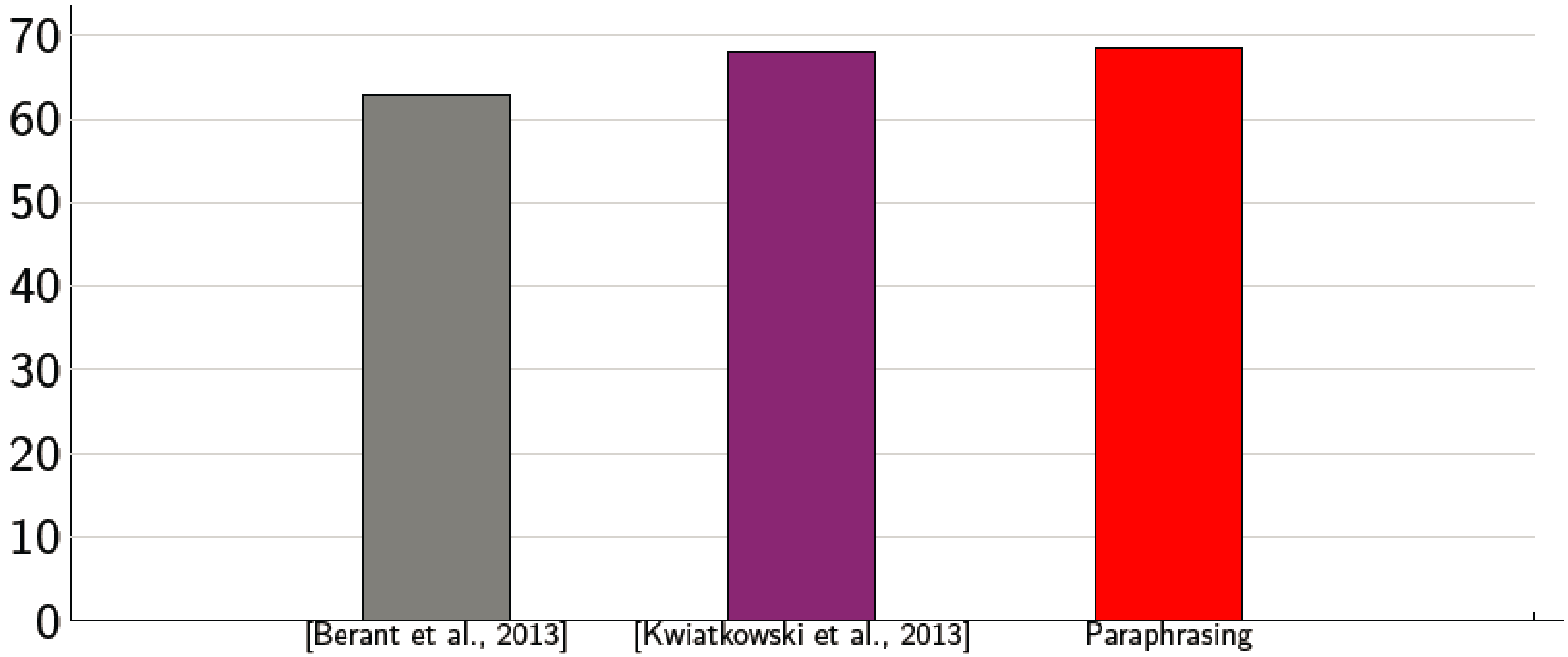
- WebQuestions dataset
  - 5810 questions
  - Crawled from Google suggest and answered using AMT
- Free917
  - 917 questions
  - Manually authored

Results on WebQuestions



Outperforms previous state-of-the-art

Results on Free917



Comparable to state-of-the-art

Question Answering with Subgraph Embeddings

Bordes et al., EMNLP 2014

Model Overview

- Learning low-dimensional vector embedding of words, Freebase entities/relation types.

- Embedding of a question close to that of its answer

- Mathematical Formulation

$$S(q,a) = f(q)^\top g(a)$$

1. **f(q)** : embedding of a question

$$f(q) = \mathbf{W}\phi(q).$$

2. **g(a):** embedding of an answer candidate

$$g(a) = \mathbf{W}\psi(a)$$

Notations

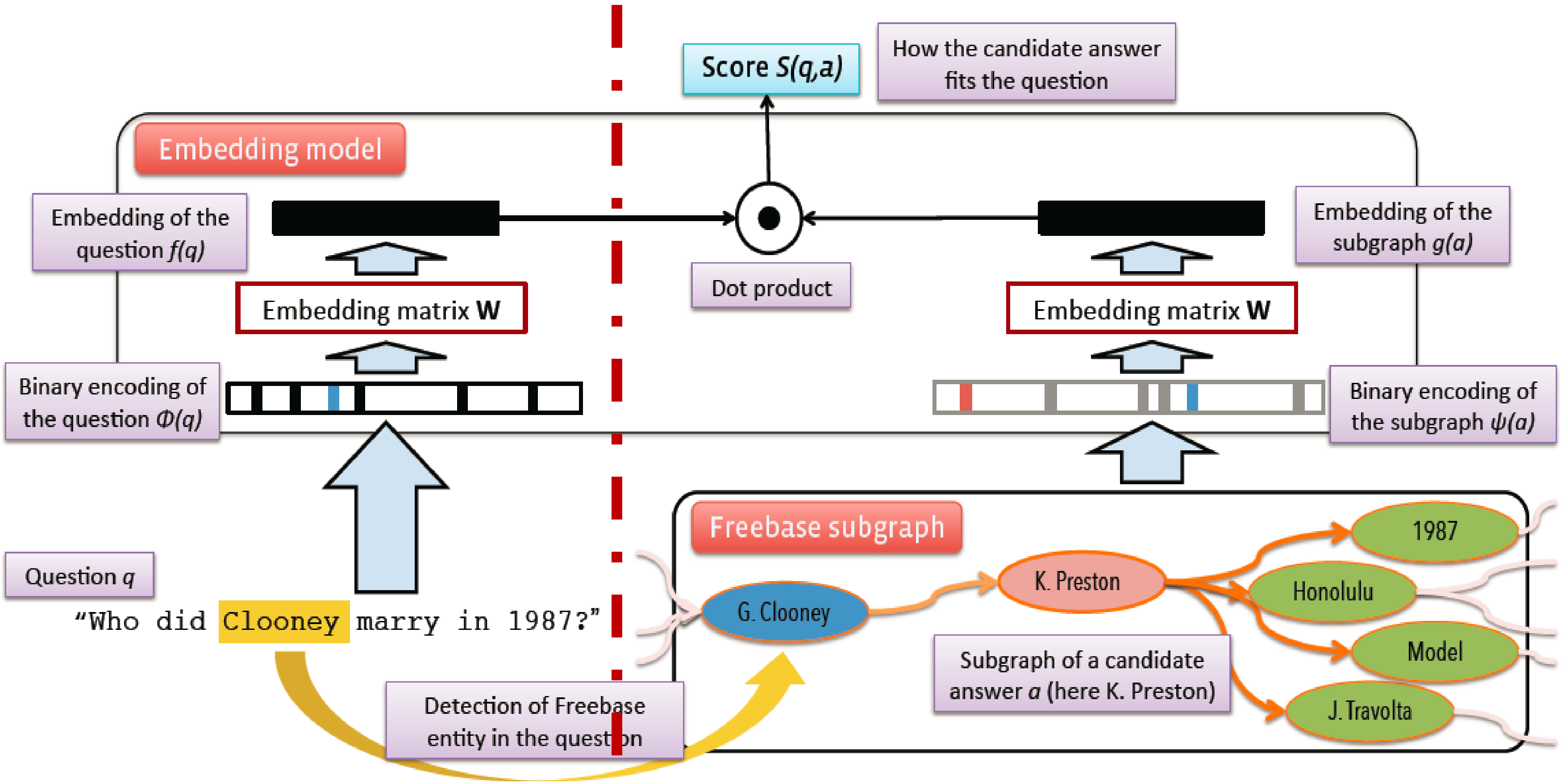
- N_W : the total number of words
- N_S : the total number of entities and relation types
- N : $N = N_W + N_S$
- *k* : the dimensionality of the embedding space

- $\phi(q) \in \mathbb{N}^N$: a sparse vector; how many times a word occurs in ***q***.

- $\psi(a) \in \mathbb{N}^N$: a sparse vector of an answer candidate **a**.

- $\mathbf{W} \in \mathbb{R}^{k \times N}$: embedding/projection matrix

Embedding Model

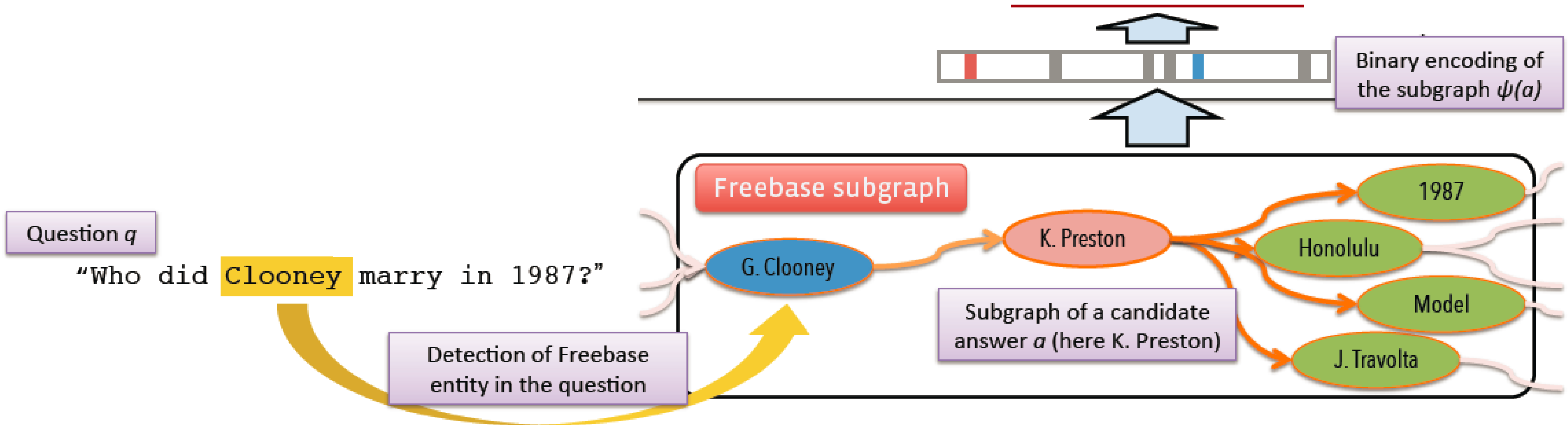


Embed a question

Embed an answer candidate

How to represent an answer candidate?

What is $\psi(a) \in \mathbb{N}^N$?



1. Locate entity in the question
2. Represent answer candidate $\psi(a)$
   a. Single entity
   b. Path: Compute 1-hop or 2-hop path from entity to answer candidate
   c. Subgraph around ***a:*** entities and relations in ***a***'s neighborhood + Path in (b)

Model

- Margin-based ranking loss function

$$\sum_{i=1}^{|\mathcal{D}|} \sum_{\bar{a} \in \bar{\mathcal{A}}(a_i)} \max\{0, m - S(q_i, a_i) + S(q_i, \bar{a})\}$$

- Training
  - Score on embedded question and answer $S(q, a) = f(q)^\top g(a)$
  - (q_i, a_i) : a question paired with its correct answer
  - $\bar{\mathcal{A}}(a_i)$: incorrect answer candidate set
    - 50% from the neighborhood of the entity in the question
    - 50% by replacing the answer entity with a random one

Inference

- Given a new question,

$$\hat{a} = \text{argmax}_{a' \in \mathcal{A}(q)} S(q, a')$$

- Answer candidate set $\mathcal{A}(q)$ construction

Let the entity in the question be ***e***,

- Strategy 1: ***C*1**
  - Entities in ***e***'s 1-hop neighborhood
- Strategy 2: ***C*2**
  - Select relation types that are most likely expressed in the question

$$S(q, a) = f(q)^{\top} g(a)$$

  - Add 2-hop entities when these relations appear in their path to ***e***

Results on WebQuestions

Method	P@1 (%)	F1 (Berant)	F1 (Yao)
Baselines			
(Berant et al., 2013)	–	31.4	–
(Bordes et al., 2014b)	31.3	29.7	31.8
(Yao and Van Durme, 2014)	–	33.0	42.0
(Berant and Liang, 2014)	–	39.9	43.0
Our approach			
Subgraph & $\mathcal{A}(q) = C_2$	**40.4**	39.2	43.2
Ensemble with (Berant & Liang, 14)	–	**41.8**	**45.7**
Variants			
Without multiple predictions	**40.4**	31.3	34.2
Subgraph & $\mathcal{A}(q) =$ All 2-hops	38.0	37.1	41.4
Subgraph & $\mathcal{A}(q) = C_1$	34.0	32.6	35.1
Path & $\mathcal{A}(q) = C_2$	36.2	35.3	38.5
Single Entity & $\mathcal{A}(q) = C_1$	25.8	16.0	17.8

Table 1: Results on the WEBQUESTIONS test set.

A Neural Network for Factoid Question Answering over Paragraphs

Iyyer et al., EMNLP 2014

Matching Texts to Entities: Quiz Bowl

- Definition
  - Given a description of an entity, identify the entity being discussed

> Later in its existence, this polity's leader was chosen by a group that included three bishops and six laymen, up from the seven who traditionally made the decision. Free imperial cities in this polity included Basel and Speyer. Dissolved in 1806, its key events included the Investiture Controversy and the Golden Bull of 1356. Led by Charles V, Frederick Barbarossa, and Otto I, for 10 points, name this polity, which ruled most of what is now Germany through the Middle Ages and rarely ruled its titular city.

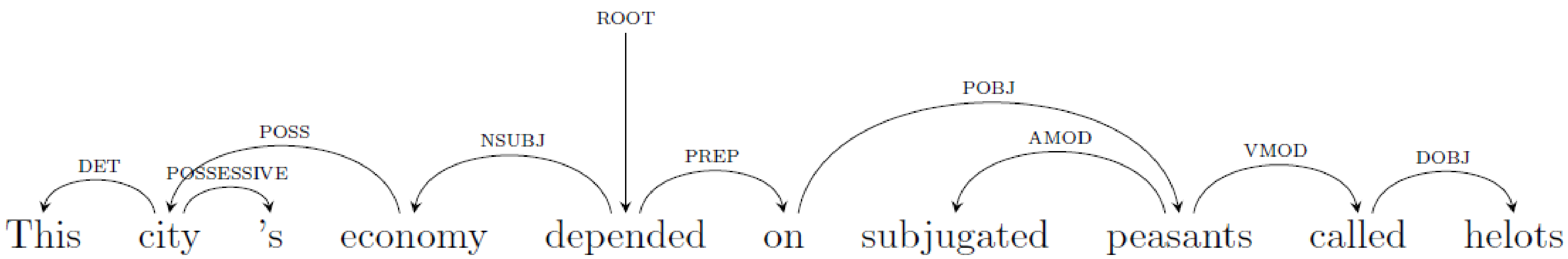
A description of " the Holy Roman Empire "

Matching Texts to Entities: Quiz Bowl

- Sentences contain hard, obscure clues
- Bag-of-words Model
- Recursive Neural Networks
  - Structure of sentences
  - Meaning of words

Dependency-Tree Recursive Neural Network

- Dependency parse tree



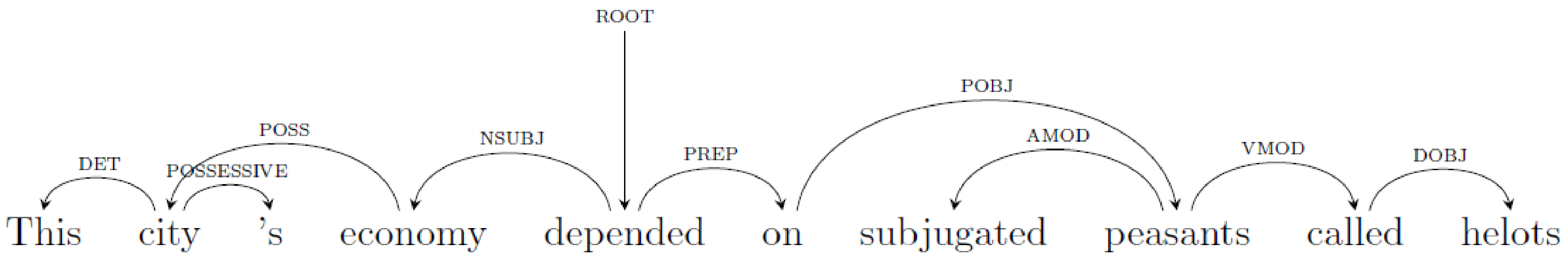
Dependency parse of a sentence from a question about Sparta.

1. Labeled directed graph
2. Nodes: lexical elements
3. Edges: dependency relations from heads to dependents
4. Existing tools: e.g., Stanford Parser

Dependency-Tree Recursive Neural Network

- Notations
  - Word representation: $x_w \in \mathbb{R}^d$.
  - Word embedding matrix W_e , $d \times V$
  - W_r : matrix corresponding to a dependency relation
  - W_v : map a word vector to a hidden representation

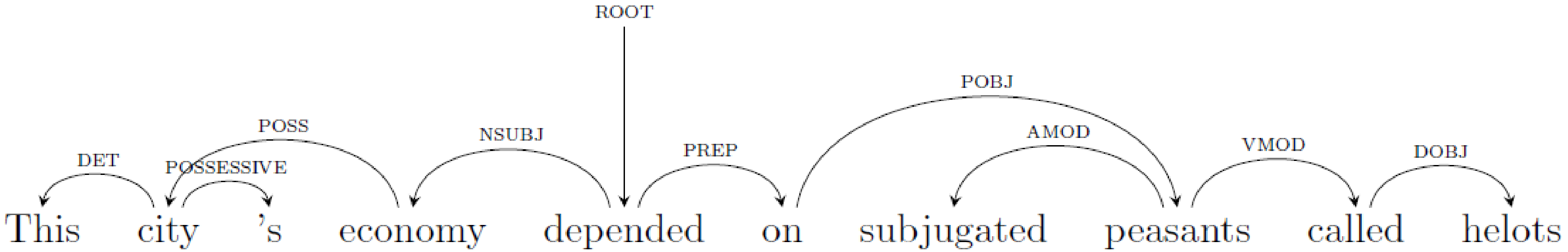
  e.g.,



$$h_{\text{helots}} = f(W_v \cdot x_{\text{helots}} + b)$$

Dependency-Tree Recursive Neural Network

- Model description



1. Hidden representation of each leaf word.

$$h_{\text{helots}} = f(W_v \cdot x_{\text{helots}} + b)$$

2. Hidden representation of its parent word.

$$h_{\text{called}} = f(W_{\text{DOBJ}} \cdot h_{\text{helots}} + W_v \cdot x_{\text{called}} + b).$$

3. Continue until the root word.

$$h_{\text{depended}} = f(W_{\text{NSUBJ}} \cdot h_{\text{economy}} + W_{\text{PREP}} \cdot h_{\text{on}} + W_v \cdot x_{\text{depended}} + b).$$

Dependency-Tree Recursive Neural Network

□ Model Formulation

Given

- a sentence S (the set of all nodes in the dependency parse tree)
- its correct answer ***c***
- its incorrect answer set ***Z***

Cost Function:

$$C(S,\theta) = \sum_{s \in S} \sum_{z \in Z} L(rank(c,s,Z)) \max(0, 1 - x_c \cdot h_s + x_z \cdot h_s),$$

where, $x_c \in W_e$, $x_z \in W_e$

$rank(c,s,Z)$ provides the rank of ***c*** w.r.t ***Z,*** and $L(r) = \sum_{i=1}^{r} 1/i.$

$\theta = (W_{r \in R}, W_v, W_e, b)$

Dependency-Tree Recursive Neural Network

- Objective function

$$J(\theta) = \frac{1}{N} \sum_{t \in T} C(t, \theta)$$

Where, ***T*** is the set of all sentences; ***N*** is the total number of nodes.

- Backpropagation

$$\frac{\partial C}{\partial \theta} = \frac{1}{N} \sum_{t \in T} \frac{\partial J(t)}{\partial \theta}$$

Dependency-Tree Recursive Neural Network

- Objective function

$$J(\theta) = \frac{1}{N} \sum_{t \in T} C(t, \theta)$$

Where, ***T*** is the set of all sentences; ***N*** is the total number of nodes.

- Backpropagation

$$\frac{\partial C}{\partial \theta} = \frac{1}{N} \sum_{t \in T} \frac{\partial J(t)}{\partial \theta}$$

- Answer prediction
  - Logistic regression classifier
  - Features: the average of all individual sentence features

Experiments

- Datasets
  - Sources

1. Publicly available quiz bowl tournament
2. NAQT : an organization that runs quiz bowl tournament

- Categories

History:

Training: 3761 questions (14217 sentences)

Testing: 699 questions (2768 sentences)

Literature:

Training: 4777 questions (17972 sentences)

Testing: 908 questions (3577 sentences)

	History			Literature		
Model	Pos 1	Pos 2	Full	Pos 1	Pos 2	Full
BOW	27.5	51.3	53.1	19.3	43.4	46.7
BOW-DT	35.4	57.7	60.2	24.4	51.8	55.7
IR-QB	37.5	65.9	71.4	27.4	54.0	61.9
FIXED-QANTA	38.3	64.4	66.2	28.9	57.7	62.3
QANTA	**47.1**	**72.1**	**73.7**	**36.4**	**68.2**	**69.1**
IR-WIKI	53.7	76.6	77.5	41.8	74.0	73.3
QANTA+IR-WIKI	**59.8**	**81.8**	**82.3**	**44.7**	**78.7**	**76.6**

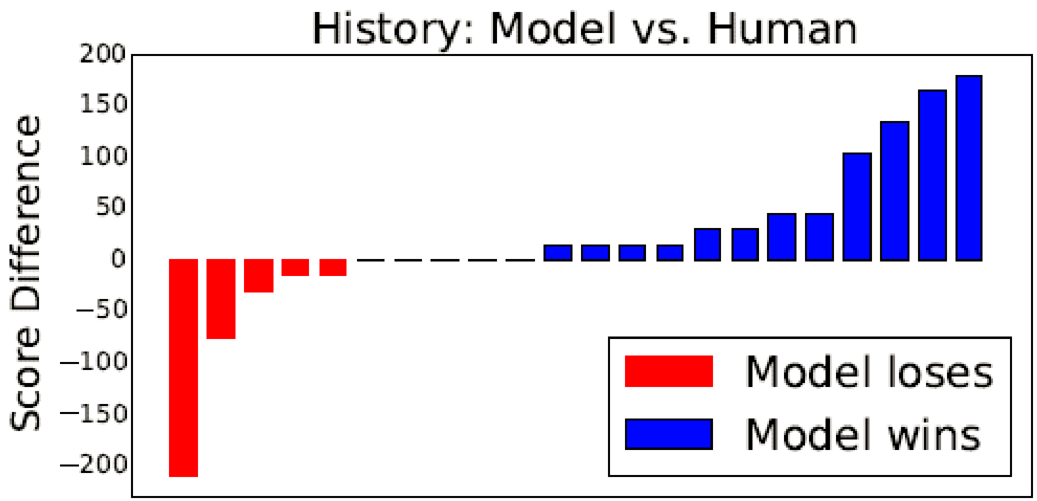
BOW: a logistic regression classifier trained on binary unigram indicators

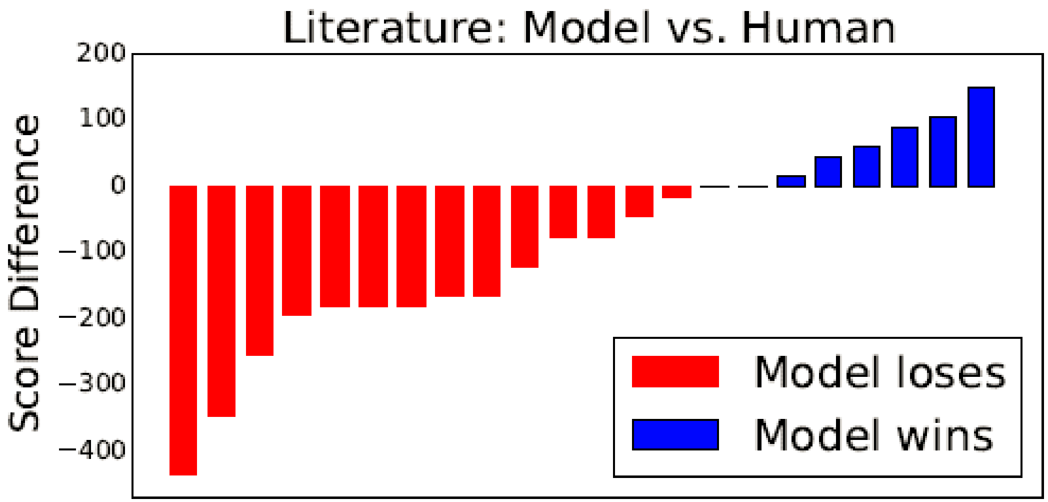
BOW-DT: BOW+ dependency relation indicator

IR-QB: Whoosh IR engine + “pages” containing training question text for each answer

IR-QB: Whoosh IR engine + “pages” containing training question text for each answer & texts from Wikipedia article

Human Comparison



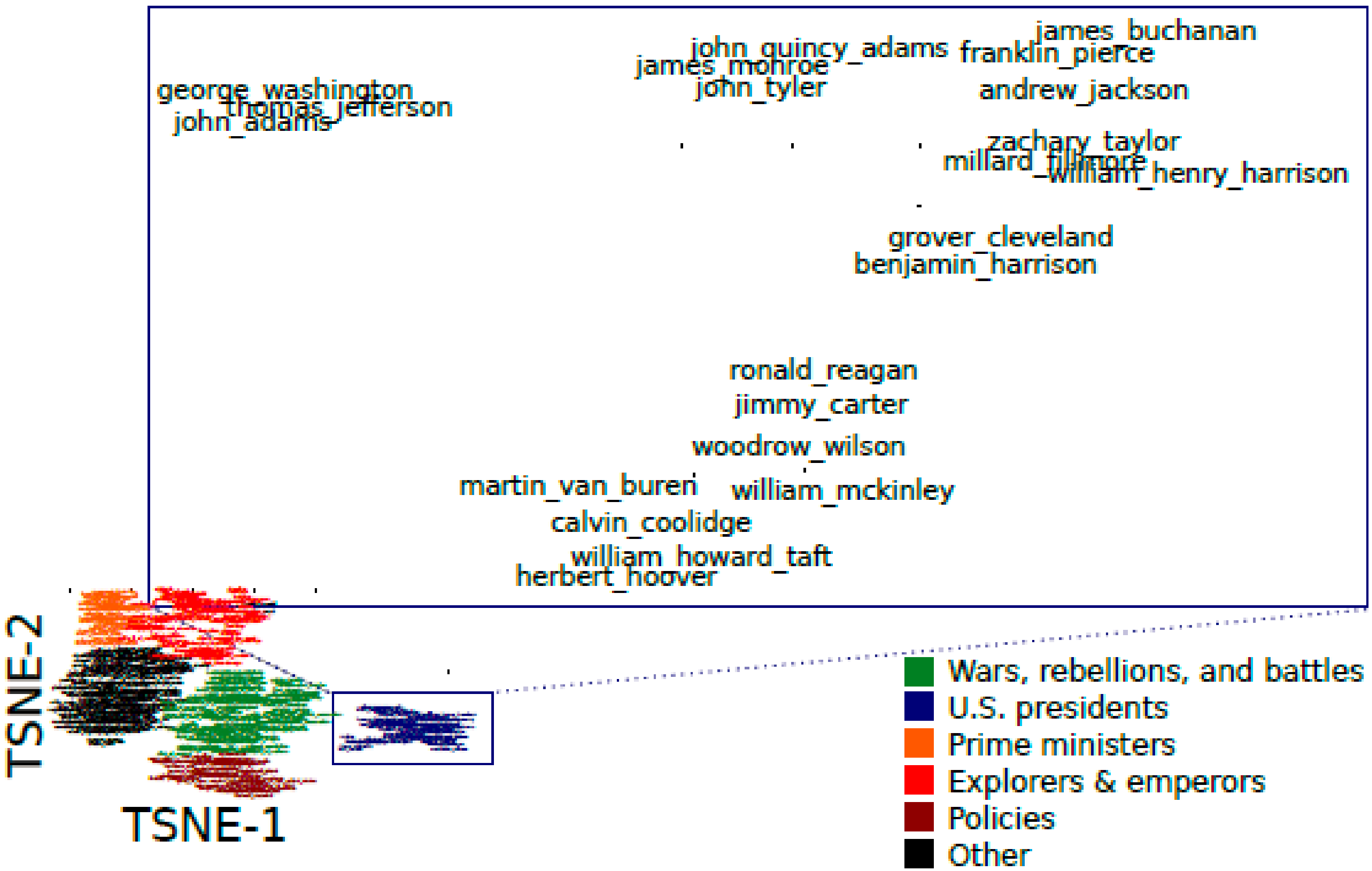


Difficult examples:

1. As a young man, this native of Genoa disguised himself as Muslim to …

2. This novel parodies freudianism in a chapter about …

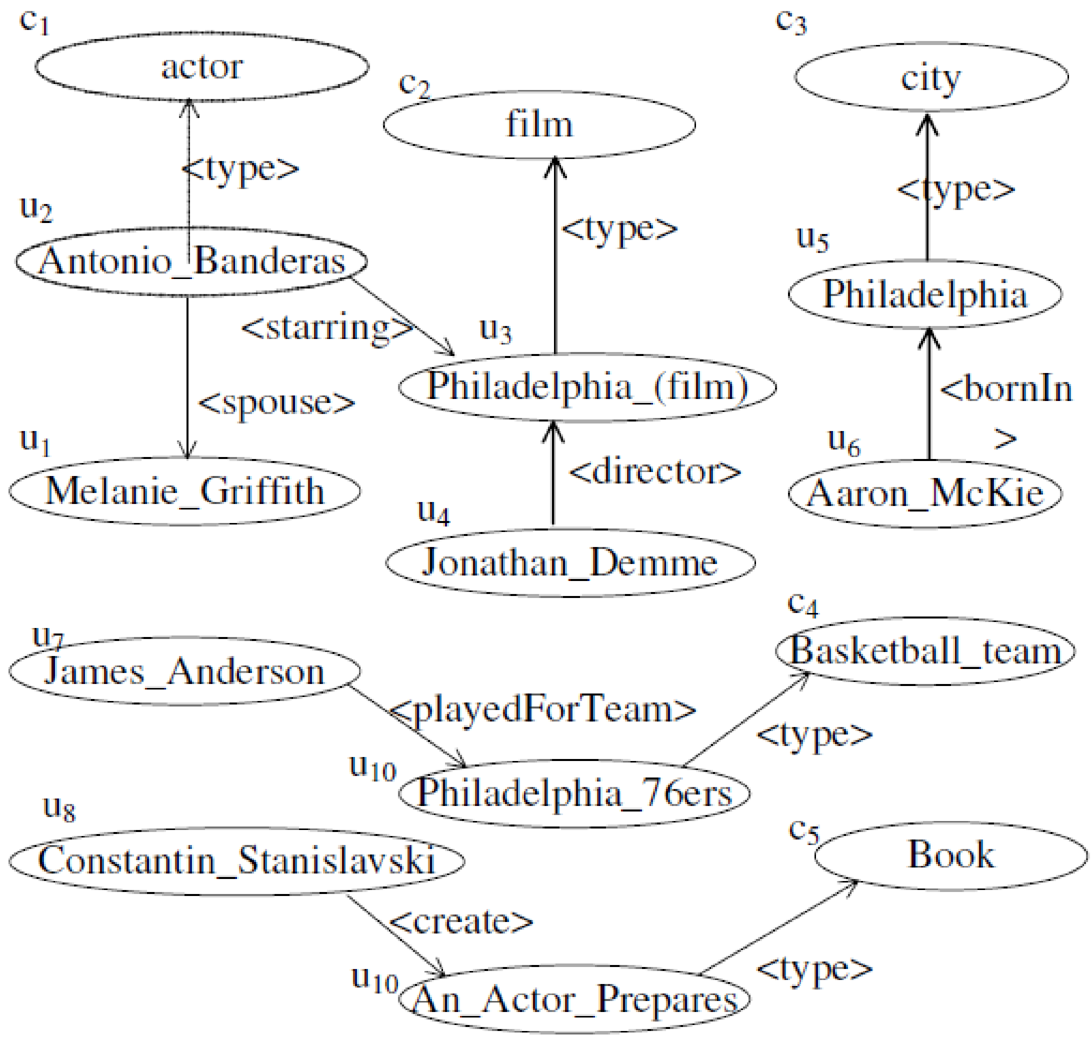
Visualization of Vectorized Answers



Natural Language Question Answering over RDF ---A Graph Data Driven Approach

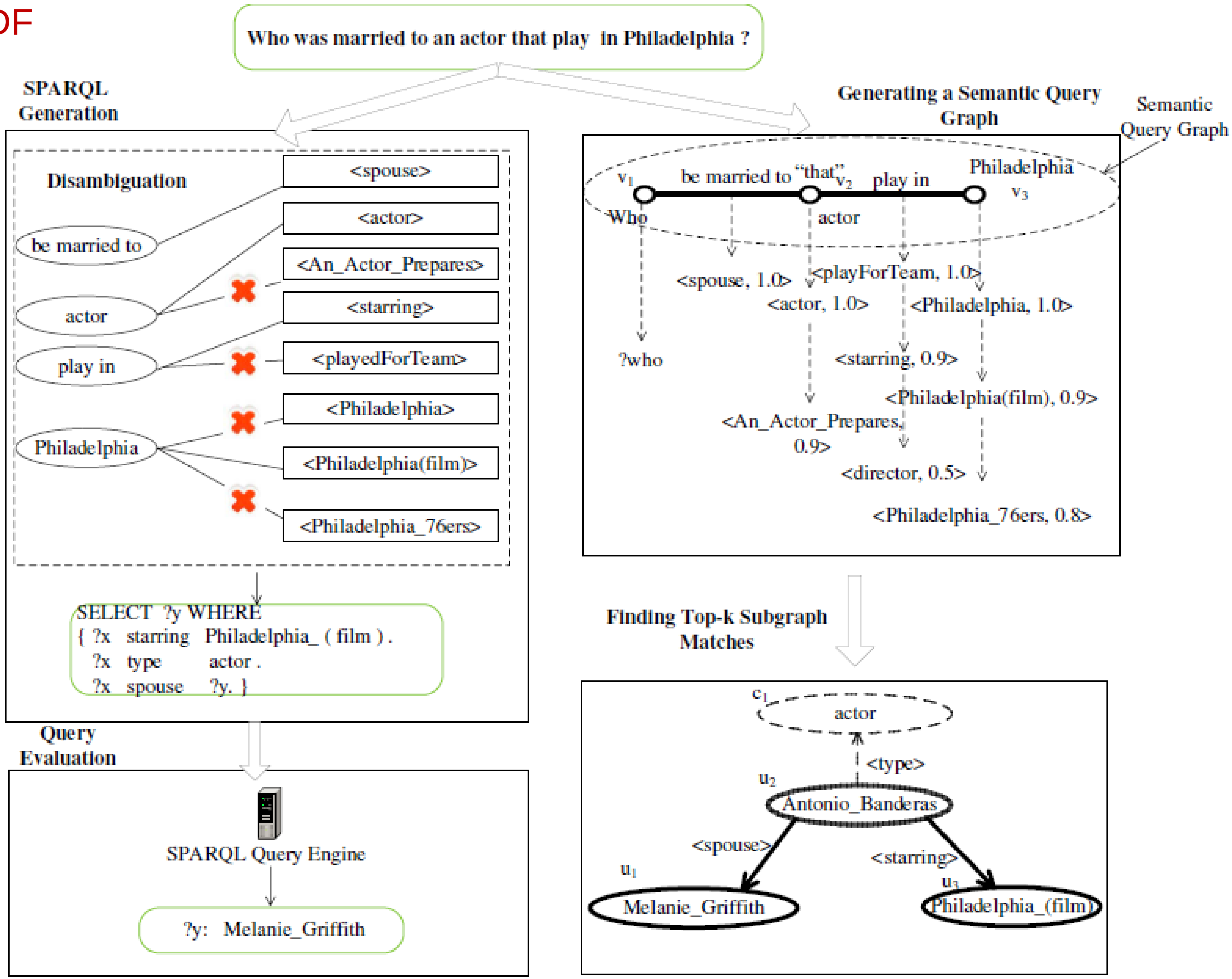
Zou et al., SIGMOD 2014

Recap: RDF Dataset and RDF Graph



Graph Representation

QA over RDF



(b) SPARQL Generation-and-Query Framework

(c) Our Framework (Zou et al., 2014)

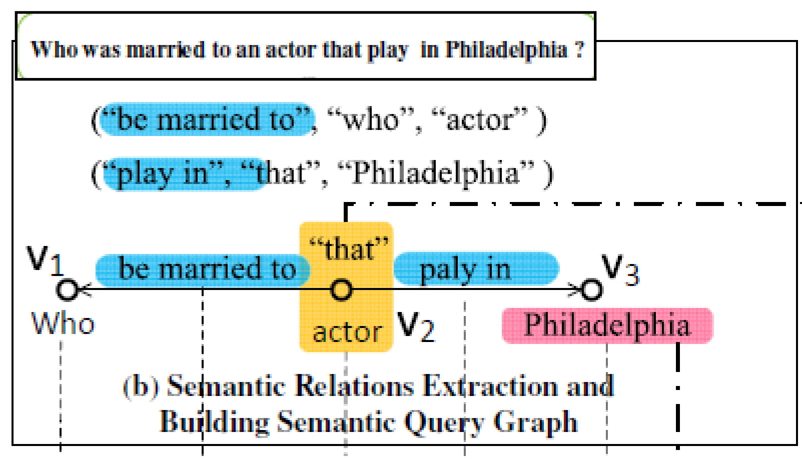
Framework

- ☐ Offline Component: to build a paraphrase dictionary

Relation Phrases	Predicates or Predicate Paths	Confidence Probability
"be married to"	<spouse>	1.0
"play in"	<starring>	0.9
"play in"	<director>	0.5
"uncle of"	<hasChild> <hasChild> <hasChild>	0.8
… …	… …	… …

Paraphrase dictionary D

- ☐ Online Component
  - Building a query graph
  - Top-K subgraph matching



(b) Semantic Relations Extraction and Building Semantic Query Graph

Offline Component

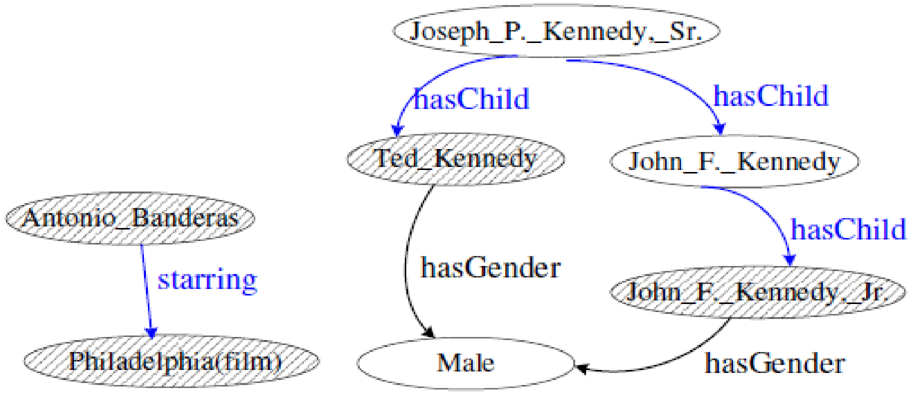
Relation Phrases ⇔ Predicates / Predicate Paths

1. Existing Systems, such as Patty and ReVerb

Table 2: Relation Phrases and Supporting Entity Pairs

Relation Phrase	Supporting Entity Pairs
"play in"	(⟨ Antonio_Banderas ⟩, ⟨ Philadelphia(film) ⟩), (⟨ Julia_Roberts ⟩, ⟨ Runaway_Bride ⟩),......
"uncle of"	(⟨ Ted_Kennedy ⟩, ⟨ John_F._Kennedy,_Jr. ⟩) (⟨ Peter_Corr ⟩, ⟨ Jim_Corr ⟩),......

2. Intuition: the relation phrase semantically equivalent to the frequent predicates or predicate paths between entity pairs



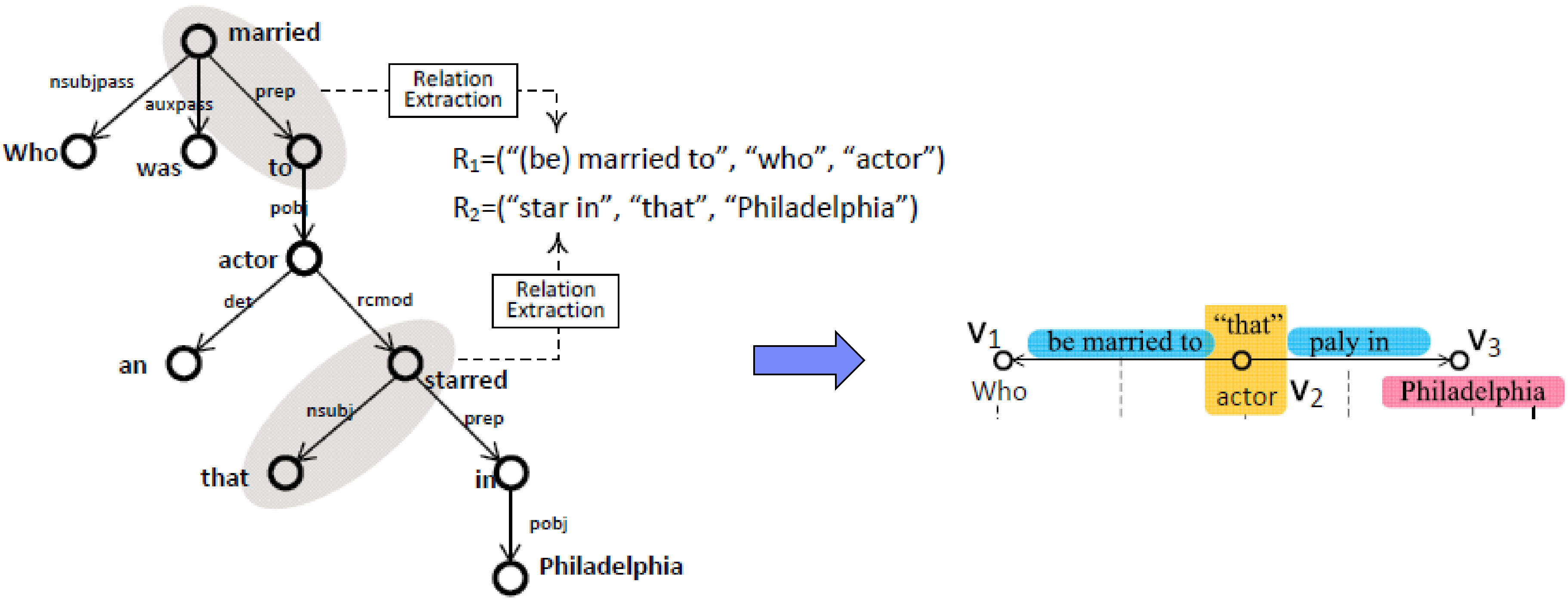
(a) "play in" (b) "uncle of"

Mapping Relation Phrases to Predicates or Predicate Paths

Online Component

Question understanding and query evaluation

1. Question understanding
   a. Dependency parsing
   b. Relation extraction, $\langle rel, arg1, arg2 \rangle$
   c. Build a semantic query graph Q^S by connecting the relations (coreference resolution)



Online Component

Question understanding and query evaluation

2. Query evaluation

a. Relation mapping, according to paraphrase dictionary D

Relation Phrases	Predicates or Predicate Paths	Confidence Probability
"be married to"	<spouse>	1.0
"play in"	<starring>	0.9

b. Vertex mapping, via entity linking (DBpedia Lookup)

Philadelphia

⟨Philadelphia⟩, ⟨Philadelphia(film)⟩ and ⟨Philadelphia_76ers⟩

c. Finding top-K subgraph matches

$$Score(M) =$$
$$\log\left(\prod_{v_i \in V(Q^S)} \delta(arg_i, u_i) \times \prod_{\overline{v_i v_j} \in E(Q^S)} \delta(rel_{\overline{v_i v_j}}, P_{ij})\right)$$

Results

Table 8: Evaluating QALD-3 Testing Questions (on DBpedia)

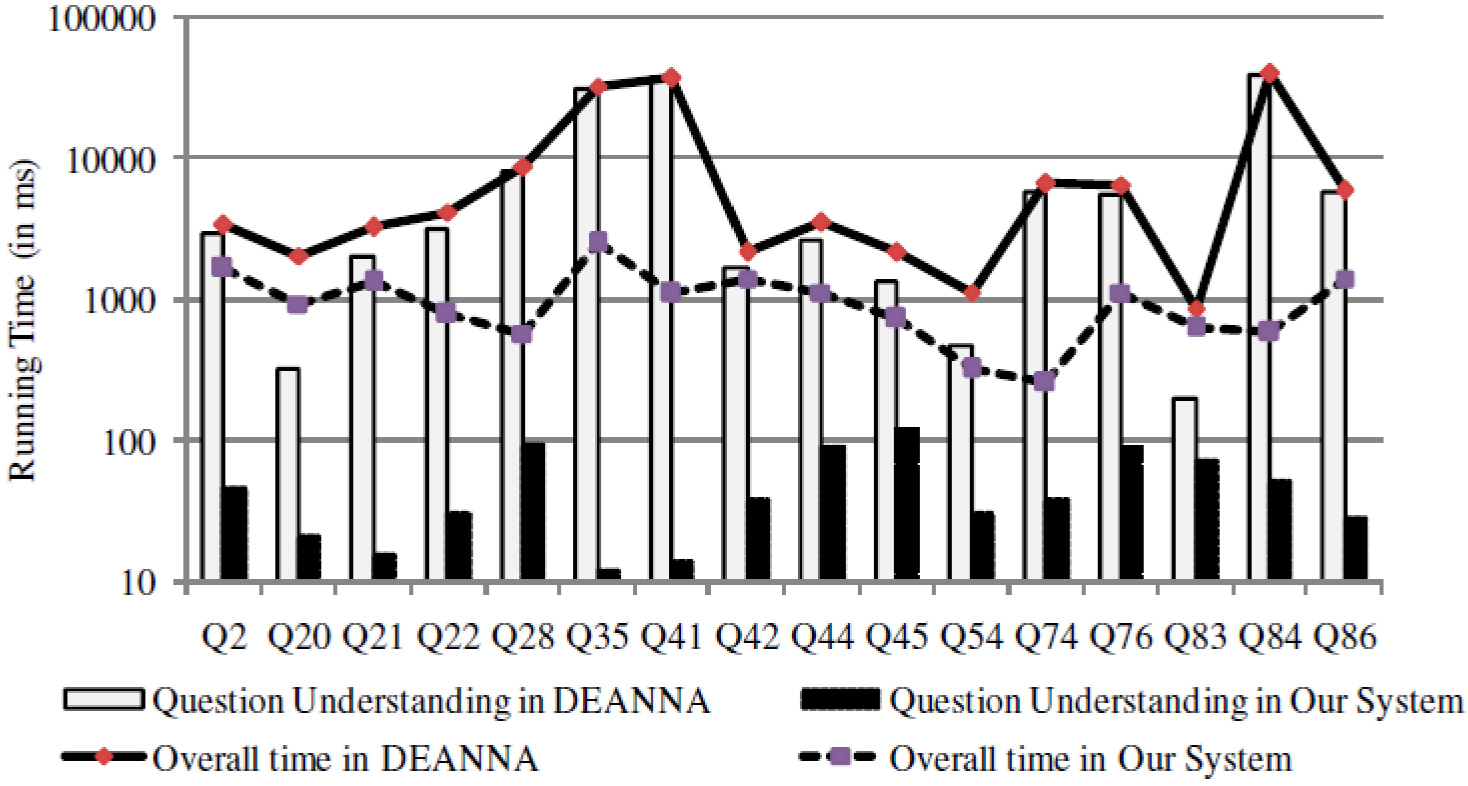
	Processed	Right	Partially	Recall	Precision	F-1
Our Method	76	32	11	0.40	0.40	0.40
squall2sparql	96	77	13	0.85	0.89	0.87
CASIA	52	29	8	0.36	0.35	0.36
Scalewelis	70	1	38	0.33	0.33	0.33
RTV	55	30	4	0.34	0.32	0.33
Intui2	99	28	4	0.32	0.32	0.32
SWIP	21	14	2	0.15	0.16	0.16
DEANNA	27	21	0	0.21	0.21	0.21

All: 99 questions

squall2sparql ***: controlled English questions*** as input

e.g., Who is the dbp: father of res:Elizabeth II?

Results



Online Running Time Comparison

faster than DEANNA by 2-68 times

Results

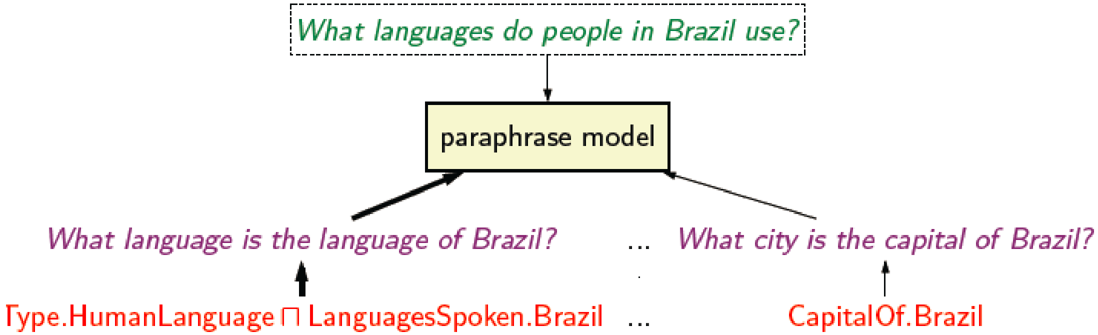
Table 10: Failure Analysis

Reason	#(Ratio)	Sample Example
Entity Linking Failure	17 (27%)	Q48: In which UK city are the headquarters of the MI6?
Relation Extraction Failure	14 (22%)	Q64. Give me all launch pads operated by NASA.
Aggregation Query	22 (35%)	Q13. Who is the youngest player in the Premier League?
Others	10 (16%)	Q37. Give me all sister cities of Brno.

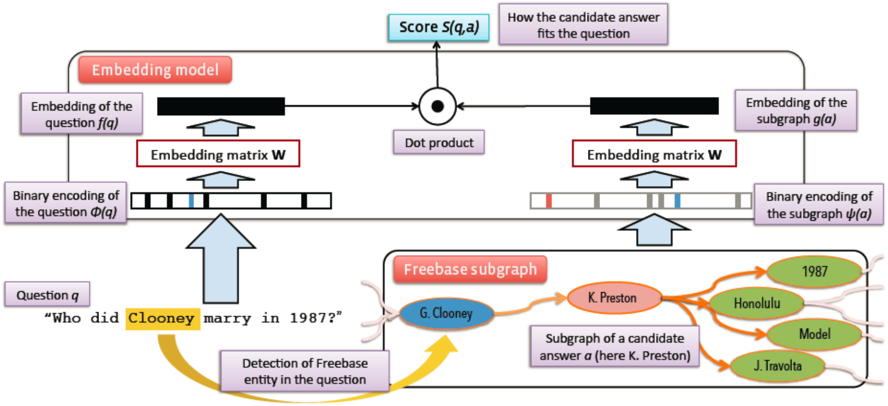
Demo Link:
http://59.108.48.18:8080/gAnswer/ganswer.jsp

Recap: Recent Methodologies for QA

- Semantic Parsing
  - Percy Liang, Stanford

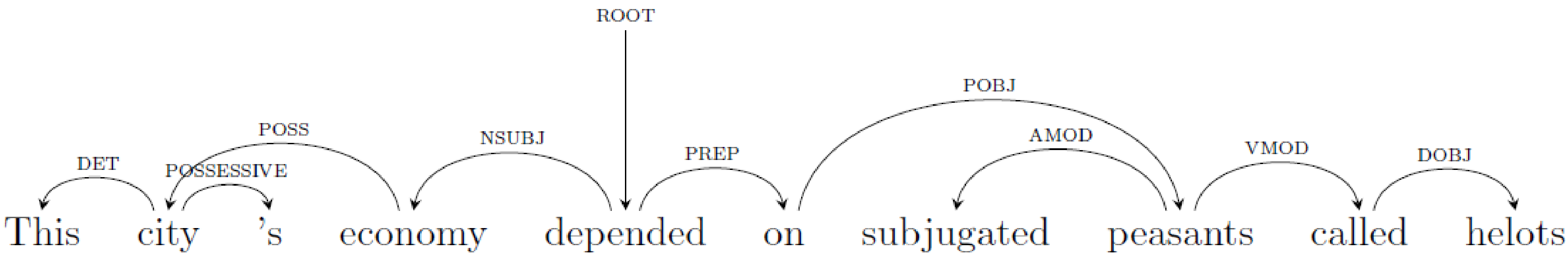


- Embedding-based
  - Jason Weston, Facebook

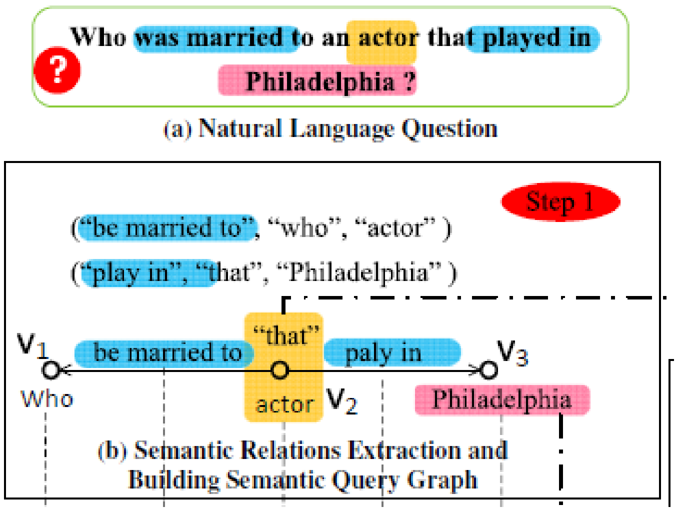


Recap: Recent Methodologies for QA

- Deep Neural Networks
  - Hal Daume III, UMD



- Graph Querying
  - Lei Zou, Peking Univ.
  - Haixun Wang, Google
  - Our group



Thank you!

Questions?

Association Model vs Vector Space Model

x : *What type of music did Richard Wagner play?*

as : *What is the musical genres of Richard Wagner?*

vs : *What composition has Richard Wagner as lyricist?*

x : *Where is made Kia car?*

as : *What place is founded by Kia motors?*

vs : *What city is Kia motors a headquarters of?*

Possible QA-related Projects

- Deep learning framework for QA
  - Convolutional Neural Networks
  - Recurrent Neural Networks (directly generate answer!)
- Incorporating semantics in resolving your task
  - Not only question answering! But general text analysis
  - Word embedding, WordNet
  - Knowledge provided in DBpedia, Freebase, YAGO etc.
- Contextual QA and dialogue systems
  - Instant feedback
  - More evidence
- Domain specific question answering
  - Discussions in all kinds of forums
  - Domain knowledge