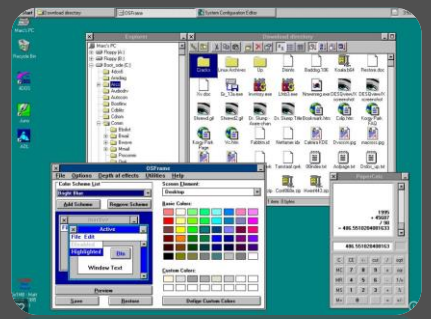
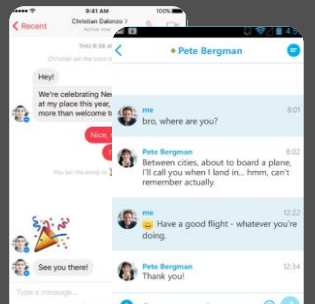
Introduction to Knowledge-based QA

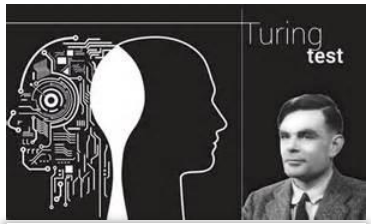
Nan Duan
Microsoft Research Asia
2018-05-16@Peking University

Microsoft

The world and technology are once again transforming.

	mid-80s	mid-90s	mid-00s	mid-10s
	PC	Web	Smartphone	Messaging
				
Platform Examples	Desktop OS Dos, Windows, Mac OS	Browser Mosaic, Explorer, Netscape	Mobile OS iOS, Android, WP	Messaging Apps Whatsapp, WeChat, Skype
Application Examples	Programs Excel, PPT, Lotus	Web Sites Yahoo, Amazon	Apps Angry Birds, Instagram	Bots / Agents Weather, Travel
UI	GUI Apps	Web Pages	Native Apps	Conversations

Evolution of Conversational AI (from 1950 to present)



Turing Test

Restricted Domain QA

Natural Language Interface to Database (NLIDB) – 1960s
- BASEBALL, 1961
- LUNAR, 1971

Reading Comprehension – 1970s
- QUALM, 1977
- …

Dialogue System – 1970s~1990s
- SHRDLU, 1972
- UNIX Consultant, 1988
- …

IBM Watson in Jeopardy!

Apple Siri

Amazon Echo

more names not in this list , due to space …

1950 | 1960s | 1970s | 1980s | 1990s | 1999-2007 | 2011 | 2014 | 2015 | 2016 | Present

Text REtrieval Conference (TREC), QA Task

TREC-8, 1999
- The first open domain QA task
- Answer document retrieval

TREC-9, 2000
- Answer span retrieval

TREC, 2003
- Answer passage retrieval

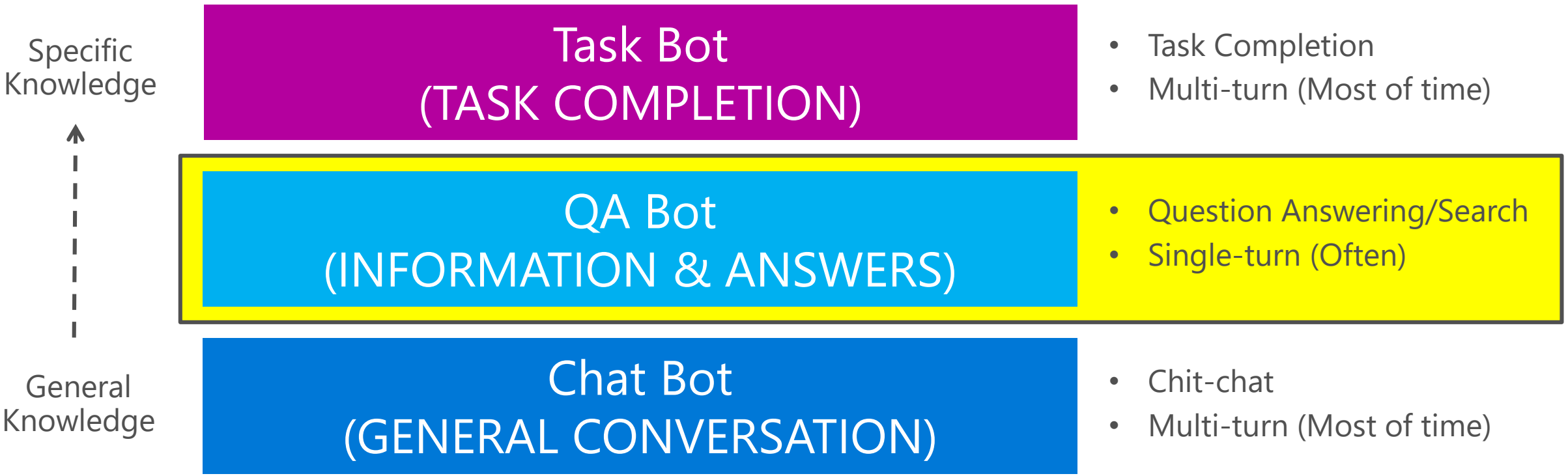
…

Microsoft Cortana & Xiaoice

Hi. I'm Cortana.

Google Home

The Logical Architecture of Conversational AI

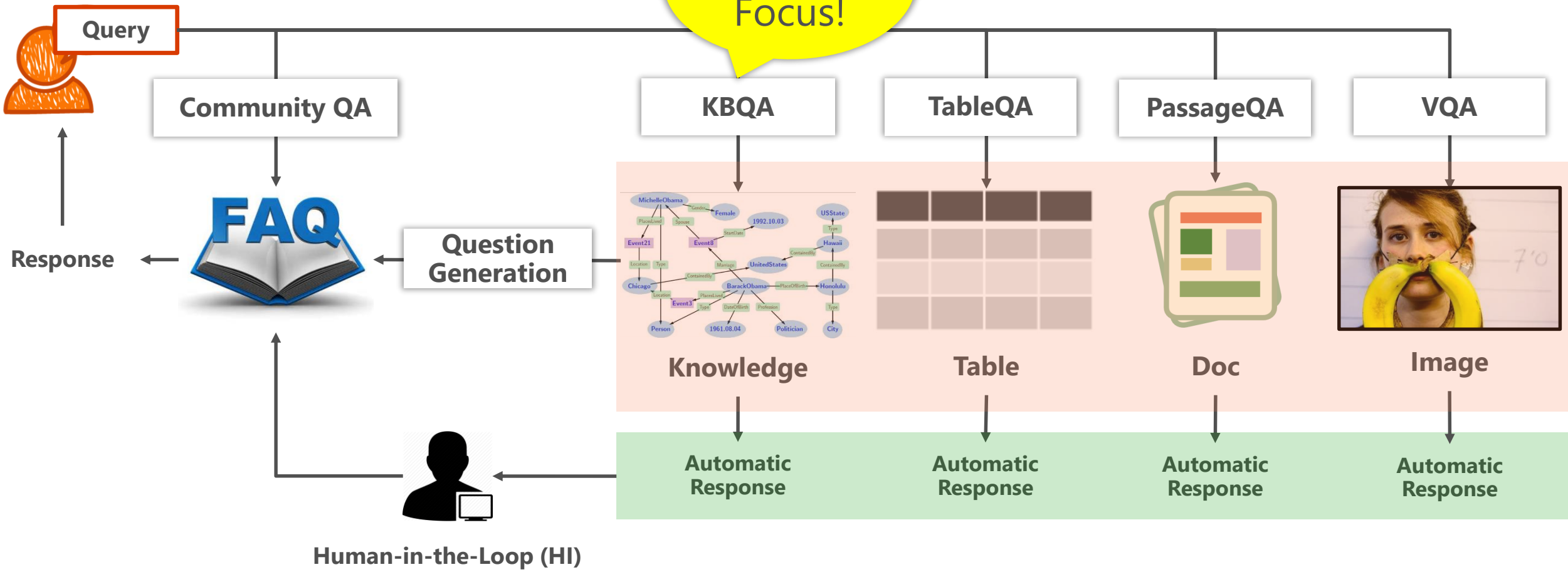


QA Bot Overview

Three core tasks

- Given a question, transform it into its logical form (**Question Understanding, QU**)
- Given a question, find its answer from existing data ... **Answering, QA**)
- Given a piece of content, predict possible quest... ...vered by the content (**Question Generation, QG**)

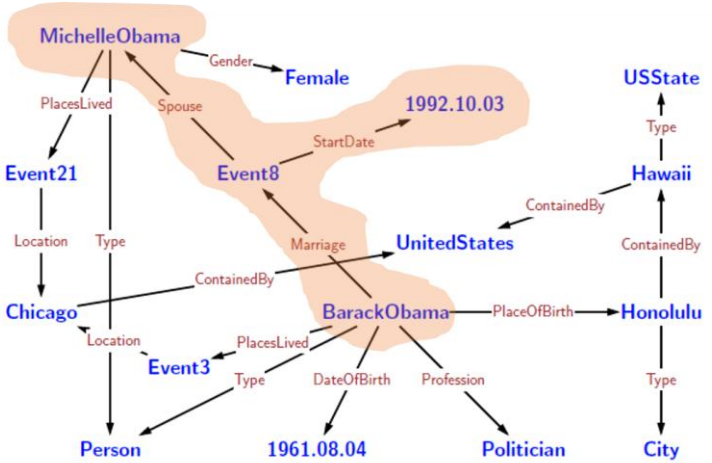
Today's Focus!



Knowledge-based QA (KBQA)

KBQA with Two Types of Knowledge

who is Michelle Obama married to in 1992 ?



Knowledge Graph

Michelle Obama

which city hosted Summer Olympic in 2008 ?

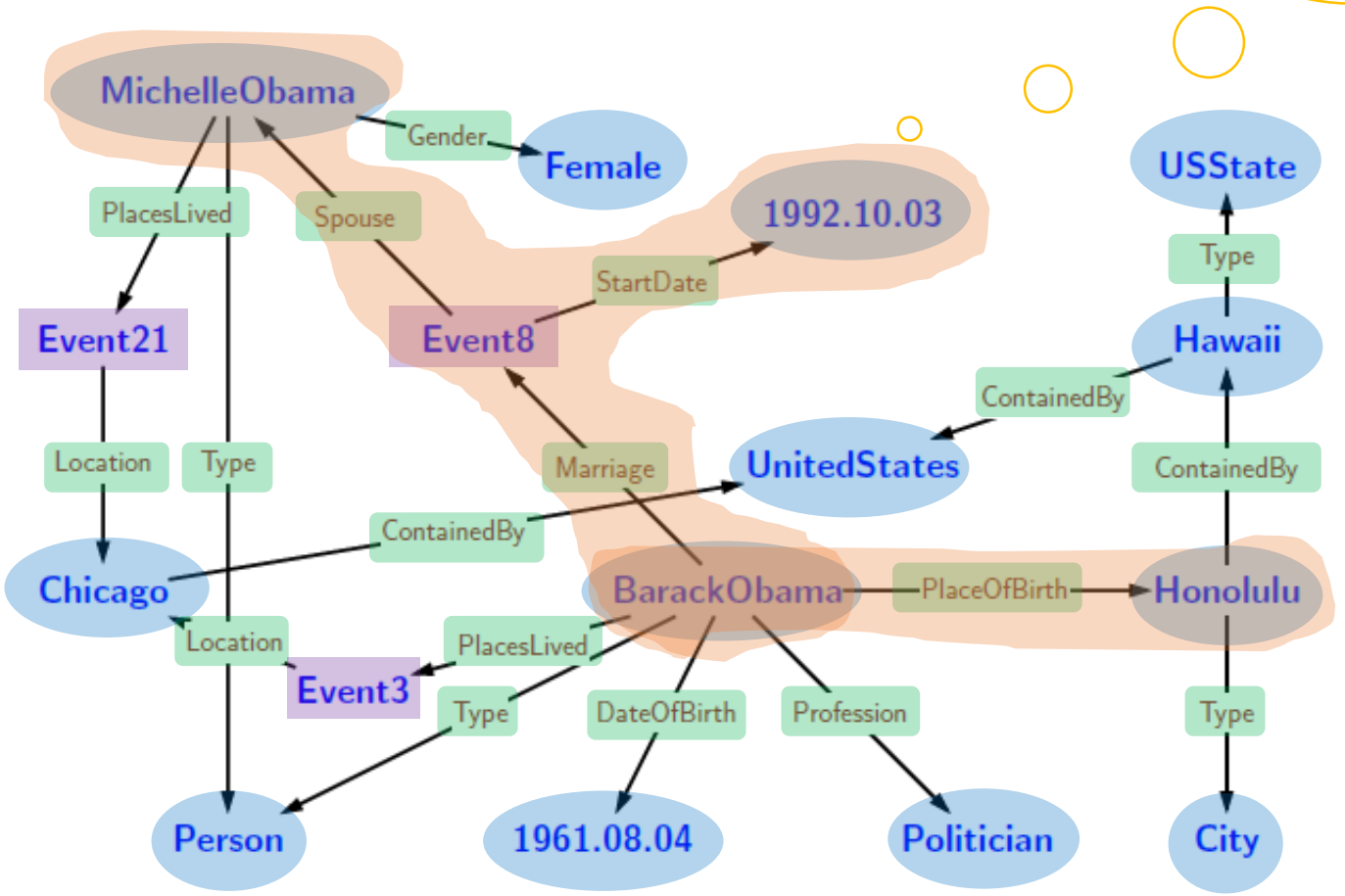
Year	City	Country	Nations
1896	Athens	Greece	14
1900	Pairs	France	24
...	...	...	...
2004	Athens	Greece	201
2008	Beijing	China	204

Semi-structured Table

Beijing

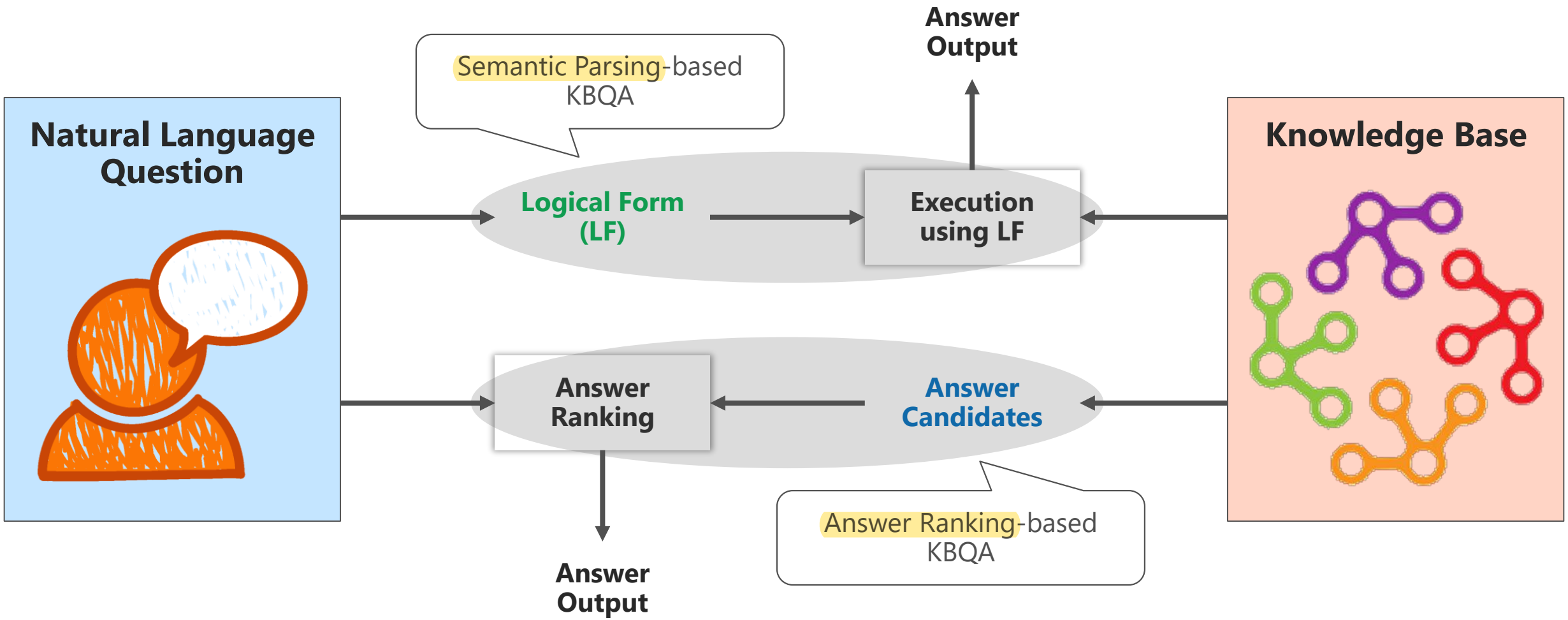
Knowledge Base (KB)

- Curated database with well-defined schema



- **Entity**
  Objects/Values in the world
- **Predicate**
  Relation between two connected entities
- **CVT** (Compound Value Type)
  Not a real-world entity, but is used to collect multiple fields of an event
- **Fact**
  Triple, which connects two entities
  Event, which connects multiple entities via a CVT node

Knowledge-based QA (KBQA): Methodology



Outline of KBQA

- Semantic Parsing-based KBQA
  1. What is **logical form**?
  2. How to **parse** a question into its logical form?
  3. How to **execute** a logical form against KB?
- Answer Ranking-based KBQA
  1. How to **select** answer candidates?
  2. How to **represent** answer candidates?
  3. How to **rank** answer candidates?

Outline of KBQA

- Semantic Parsing-based KBQA
  1. What is **logical form**?
  2. How to **parse** a question into its logical form?
  3. How to **execute** a logical form against KB?
- Answer Ranking-based KBQA
  1. How to **select** answer candidates?
  2. How to **represent** answer candidates?
  3. How to **rank** answer candidates?

Lambda Calculus (λ-Calculus) as Logical Form (LF)

- λ-Calculus was introduced by Alonzo Church in 1930s
- Any computable function can be expressed using this formalism
- The core concept in λ-Calculus is "expression"
- An **expression** is defined recursively as follows

⟨expression⟩ ≔ ⟨constant⟩ | ⟨variable⟩ | ⟨function⟩ | ⟨application⟩

⟨function⟩ ≔ λ ⟨variable⟩. ⟨expression⟩

⟨application⟩ ≔ ⟨expression⟩ ⟨expression⟩

Lambda Calculus: Constant

- Represent objects in the world

China, Bill Gates, Mount Everest, 2017, …

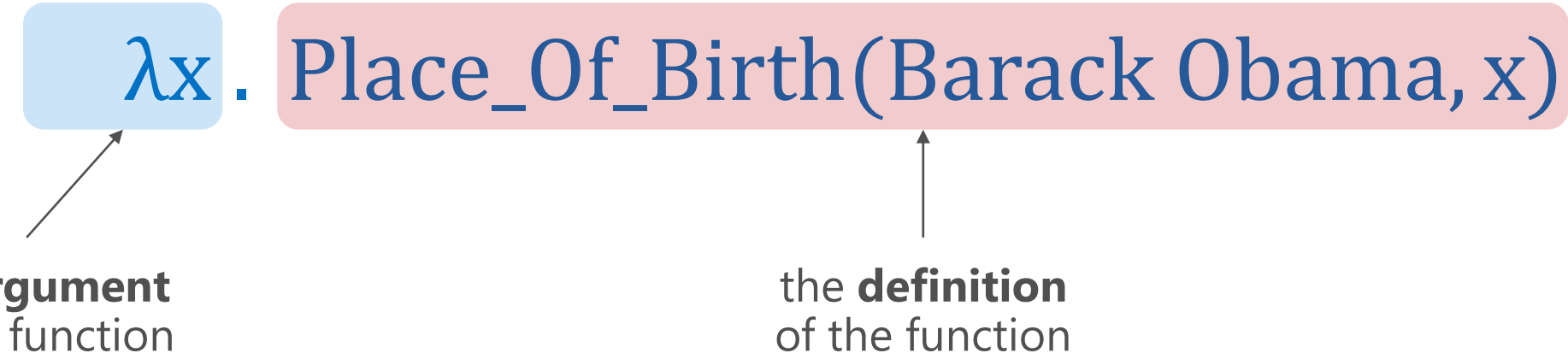
Lambda Calculus: Variable

- Represent object variables

$$x, y, z, \ldots$$

Lambda Calculus: Function

- Represent a function, and return the output of the function



Lambda Calculus: Application

- Apply the first expression to the second expression

$$\lambda x \lambda y.\, \text{Place\_Of\_Birth}(x, y) \quad \lambda x.\, (x = \text{Barack Obama})$$

$$\lambda y.\, \text{Place\_Of\_Birth}(\text{Barack Obama}, y)$$

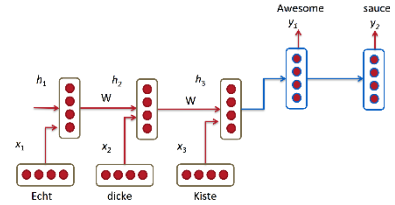
Transforming Natural Language into Logical Form (λ-Calculus)

Natural Language

What city was Obama born ?

Semantic Parsing

Grammar-based Semantic Parsing



Neural Network-based Semantic Parsing

Logical Form

$$\lambda x.\, \mathrm{Type}(\mathrm{City}, x) \wedge \mathrm{Place\_of\_Birth}(\mathrm{Barack\ Obama}, x)$$

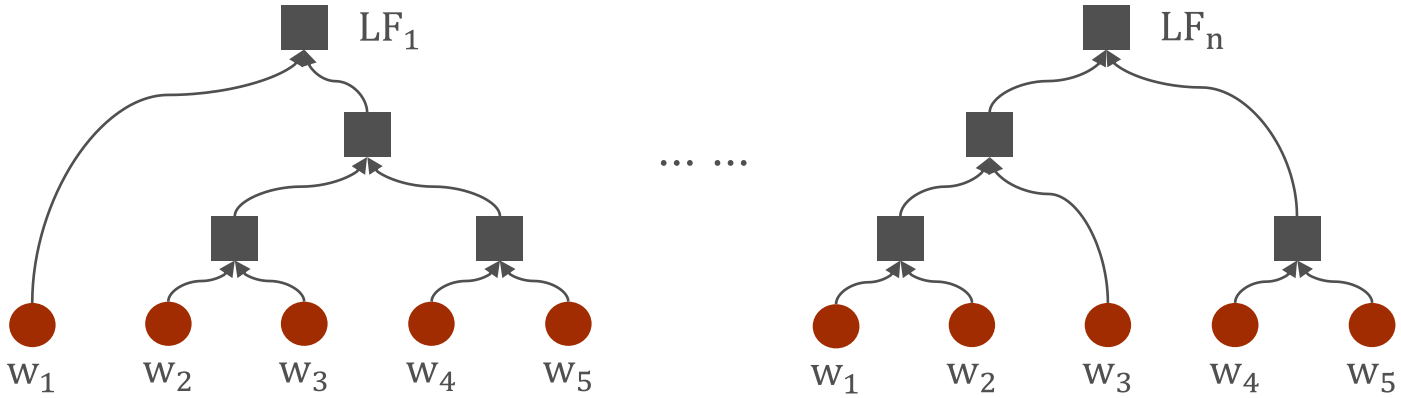
Generic Framework of Grammar-based Semantic Parsing

$$\widetilde{\mathrm{LF}} = \underset{\mathrm{LF}\in\{\mathrm{LF}_1,\ldots,\mathrm{LF}_n\}}{\operatorname{argmax}} \mathrm{P}(\mathrm{LF}|\mathrm{Q};\bar{\theta}) = \underset{\mathrm{LF}\in\{\mathrm{LF}_1,\ldots,\mathrm{LF}_n\}}{\operatorname{argmax}} \sum_{\mathrm{D}} \mathrm{P}(\mathrm{LF},\mathrm{D}|\mathrm{Q};\bar{\theta})$$

Logical Form Ranking

Rank different logical forms

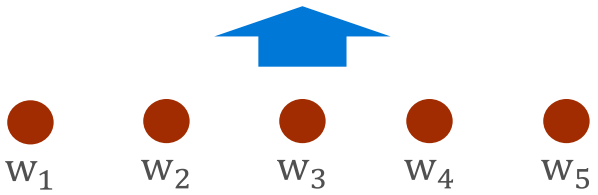
- Q: a question
- LF: a logical form of Q
- D: a semantic parsing derivation
- $\bar{\theta}$: model parameters



Semantic Rule-Driven CYK Parsing

Parse Q into possible logical forms $\{\mathrm{LF}_1, \ldots, \mathrm{LF}_n\}$ using semantic rules

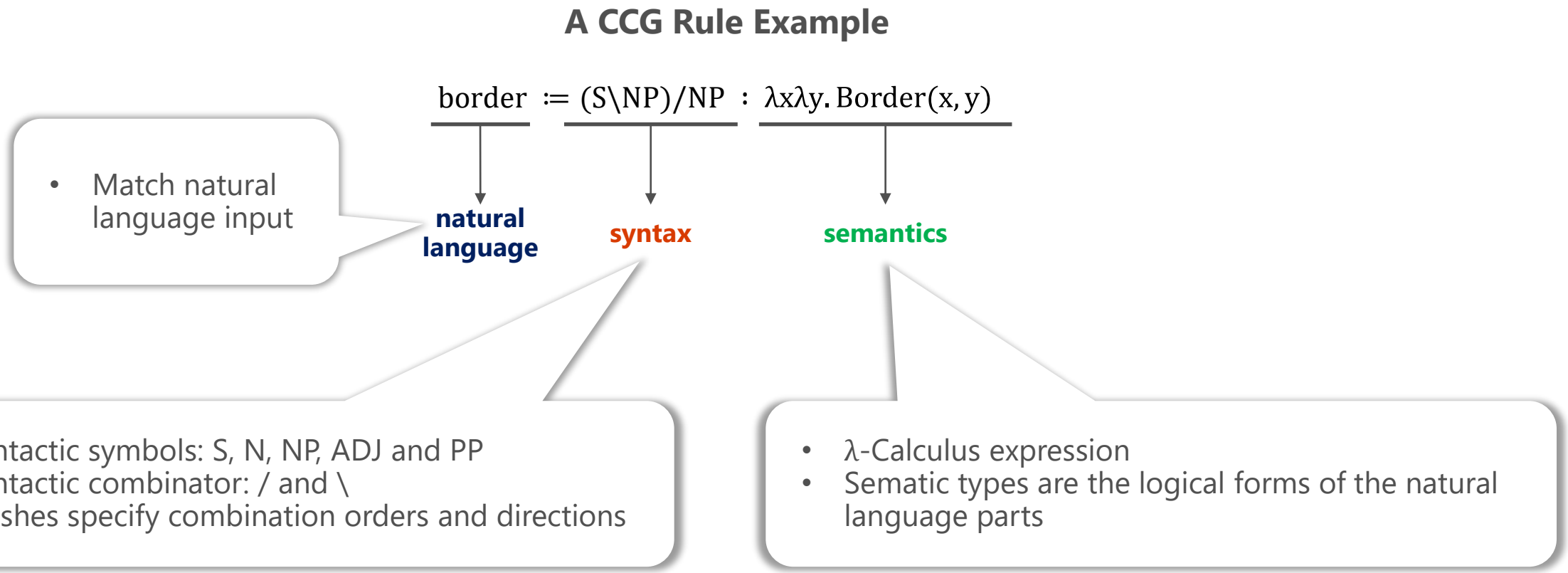
- CCG
- SCFG
- λ-DCS
- …



Natural Language Question

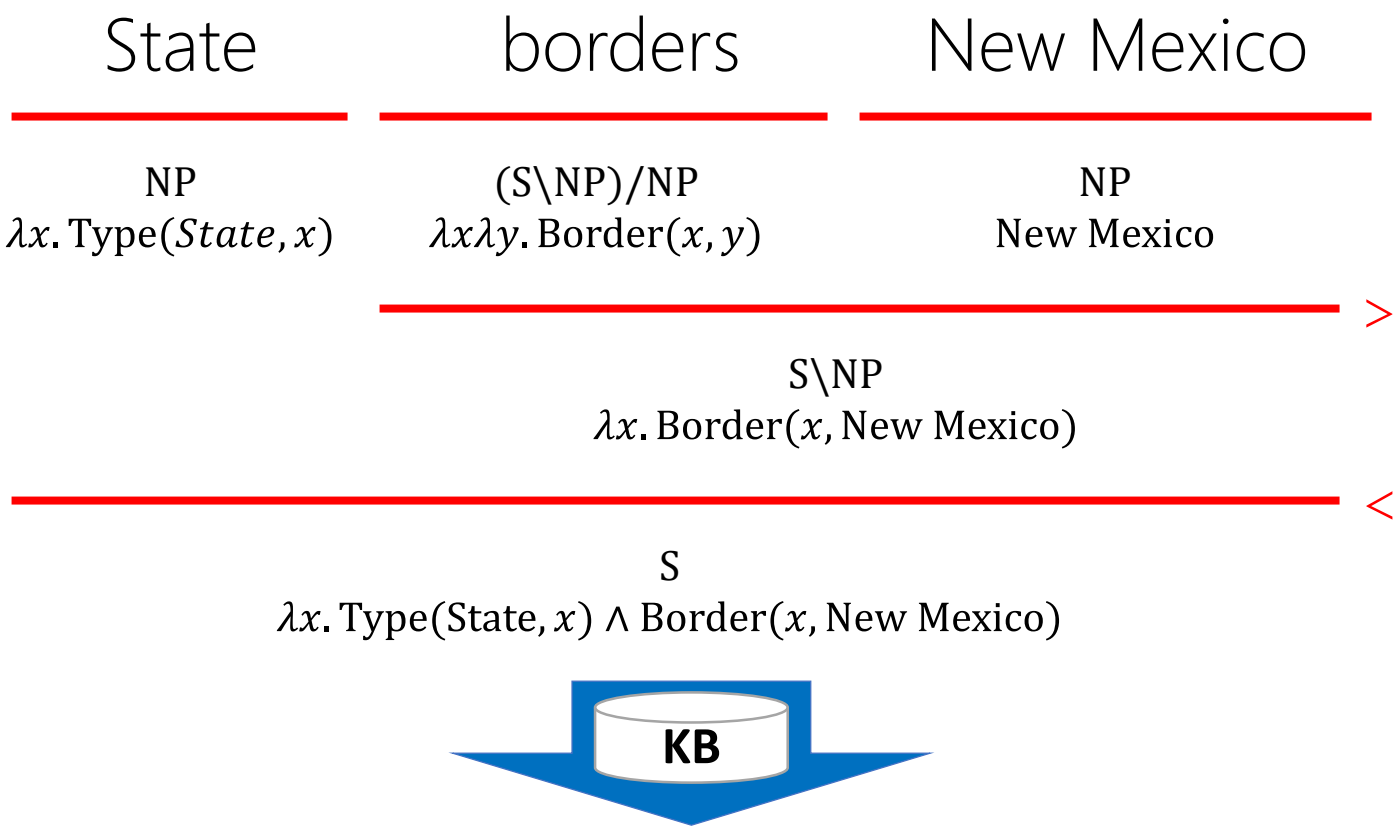
Combinatorial Categorial Grammar (CCG)

- CCG captures syntactic and semantic information jointly



Semantic Parsing with CCG

(Zettlemoyer and Collins, 2007; Kwiatkowski et al., 2011)



Arizona, Colorado, Oklahoma, Texas

CCG Rule Mining

- Input (<question, logical form> pairs)

Texas borders New Mexico
borders(texas, new_mexico)

use rules to extract all possible <Q, LF> pairs

Category Rules

Input Trigger	Output Category
constant c	NP : c
arity one predicate p	N : λx.p(x)
arity one predicate p	S \ NP : λx.p(x)
arity two predicate p	(S \ NP) / NP : λx.λy.p(y, x)
arity two predicate p	(S \ NP) / NP : λx.λy.p(x, y)
arity one predicate p	N / N : λg.λx.p(x) ∧ g(x)
arity two predicate p and constant c	N / N : λg.λx.p(x, c) ∧ g(x)
arity two predicate p	(N \ N) / NP : λx.λg.λy.p(y, x) ∧ g(x)
arity one function f	NP / N : λg.argmax/min(g(x), λx.f(x))
arity one function f	S / NP : λx.f(x)

- Output (CCG rules)

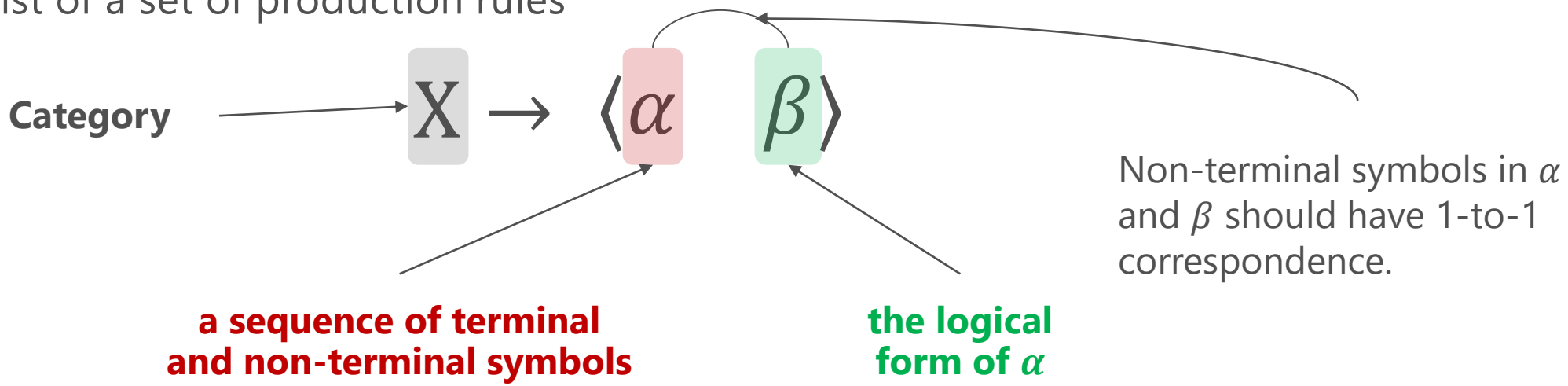
Texas := NP : texas
borders := (S \ NP) / NP : λx.λy.borders(y, x)
New Mexico := NP : new_mexico

1. **maximize the likelihood:** $\prod_i P_w(LF_i|Q_i) = \prod_i \sum_d P_w(LF_i, d|Q_i)$

2. **keep CCG rules that occur in the highest scoring derivations of training data**

Synchronous Context Free Grammar (SCFG)

- SCFG captures lexical and semantic information jointly
- A SCFG consist of a set of production rules



- Examples

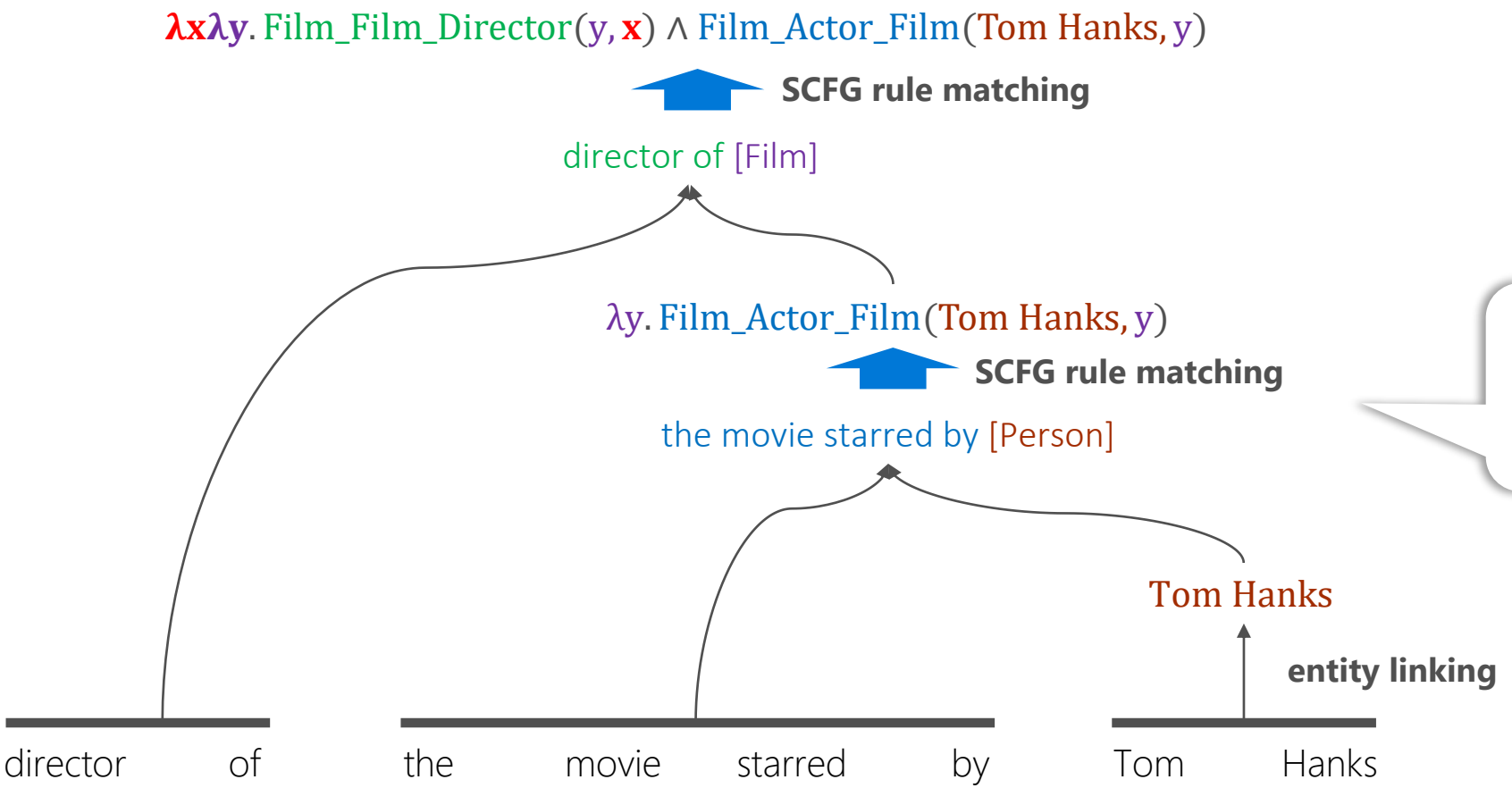
$[\text{Person}] \rightarrow \langle \text{Tom Hanks} \quad \text{Tom Hanks} \rangle$

$[\text{Film}] \rightarrow \langle \text{the moive starred by } [\text{Person}]_1 \quad \lambda x.\, \text{Film\_Actor\_Film}([\text{Person}]_1, x) \rangle$

$[\text{Film}] \rightarrow \langle \text{the moive starred by Tom Hanks} \quad \lambda x.\, \text{Film\_Actor\_Film}(\text{Tom Hanks}, x) \rangle$

Semantic Parsing with SCFG

(Bao et al., 2014; Wong and Mooney, 2007)



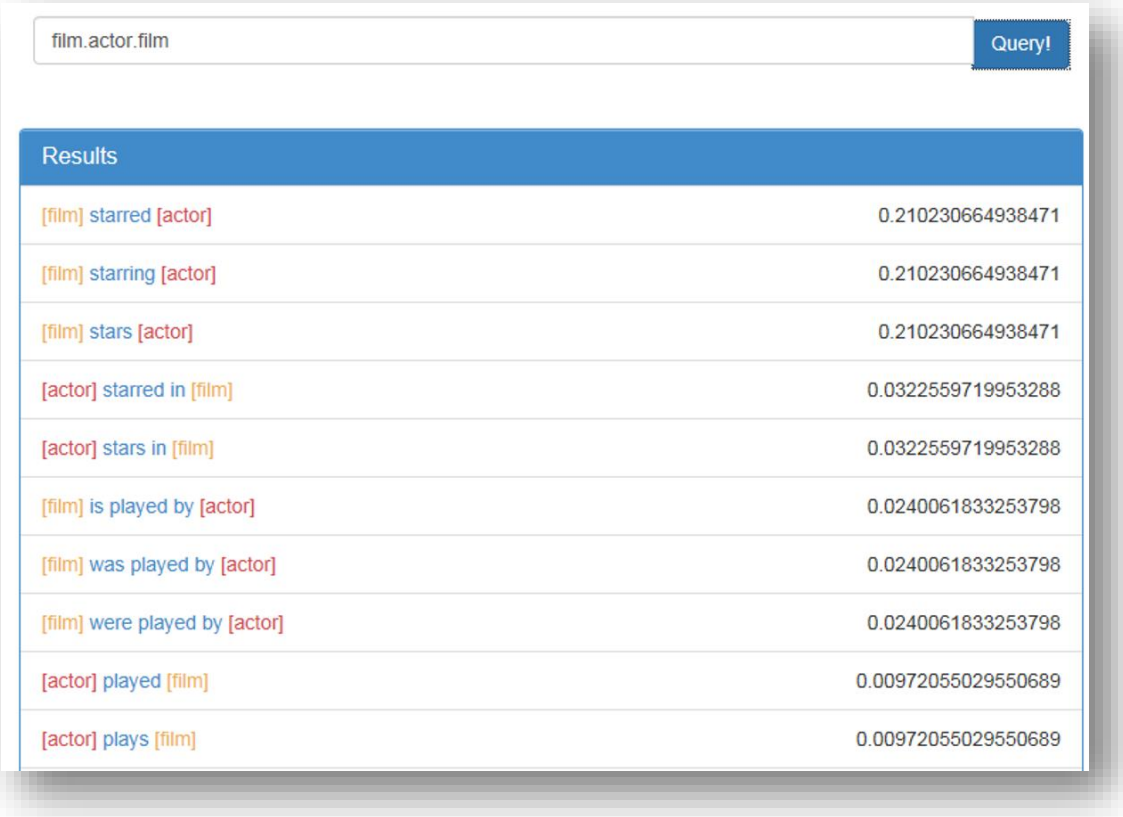
Lots of semantic derivations will be generated during this procedure.

SCFG Rule Mining

Paired Entities of a given KB Predicate

Film.Film.Director

<Forrest Gump, Robert Zemeckis>
<Titanic, James Cameron>
<Rain Man, Barry Levinson>
…



Passage Retrieval from Raw Text

Robert Zemeckis is director of Forrest Gump

Titanic was a movie directed by James Cameron

Barry Levinson was famous as the director of Rain Man

Relation Patterns

Film.Film.Director

[Director] *is director of* **[Film]**	0.84
[Film] *was a movie directed by* **[Director]**	0.81
[Director] *was famous as the director of* **[Film]**	0.77
…	…

→ [Director] → ⟨is director of $[\text{Film}]_1$ $\lambda x.\ \text{Film\_Director\_Film}(x, [\text{Film}]_1)$⟩

Lambda Dependency-based Compositional Semantics (λ-DCS)

- λ-DCS is another formal language
- λ-DCS attempts to remove explicit use of variables, so it is simpler than λ-Calculus
- λ-DCS is specially designed for Freebase Knowledge Graph

Question	people who have lived in Seattle ?
λ-Calculus	$\lambda x.\, \exists y.\, \mathrm{PlaceLived}(x, y) \wedge \mathrm{Location}(y, \mathrm{Seattle})$
λ-DCS	PlaceLived. Location. Seattle

Building Blocks of λ-DCS

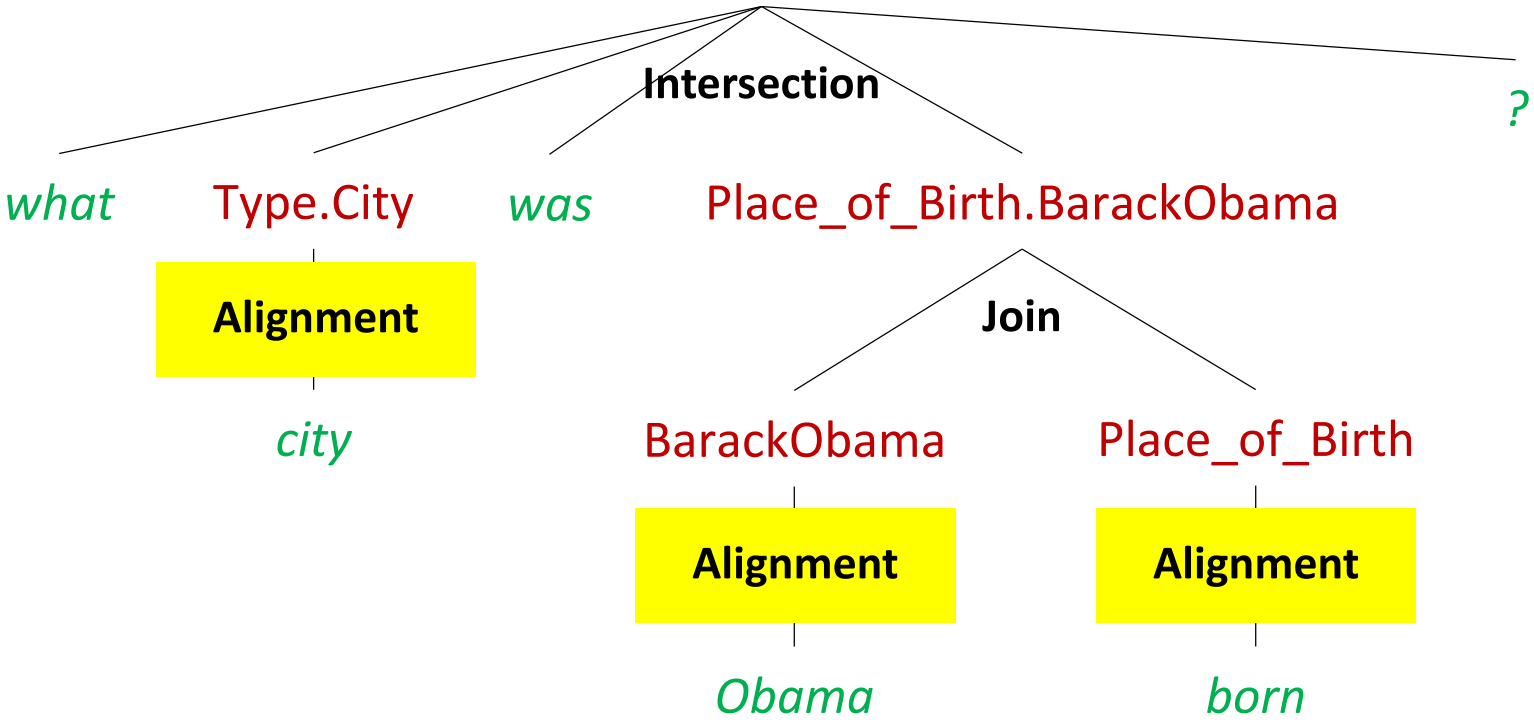
- **Entity**
  - NL: Seattle
  - LF: Seattle $\leftrightarrow \lambda x.\,[x = \text{Seattle}]$
- **Predicate**
  - NL: birthday
  - LF: PlaceOfBirth $\leftrightarrow \lambda x \lambda y.\,\text{PlaceOfBirth}(x, y)$
- **Join Operator (.)**
  - NL: people who was born in Seattle
  - LF: PlaceOfBirth. Seattle $\leftrightarrow \lambda x.\,\text{PlaceOfBirth}(x, \text{Seattle})$
- **Intersection Operator (⊓)**
  - NL: people who are scientist and born in Seattle
  - LF: Profession. Scientist ⊓ PlaceOfBirth. Seattle $\leftrightarrow \lambda x.\,\text{Profession}(x, \text{Scientist}) \wedge \text{PlaceOfBirth}(x, \text{Seattle})$
- **Union Operator (⊔)**
  - NL: Movie directed or played by Tom Hanks
  - LF: Directed_By. TomHanks ⊔ Starred_By. TomHanks $\leftrightarrow \lambda x.\,\text{Directed\_By}(x, \text{Tom Hanks}) \vee \text{Starred\_By}(x, \text{TomHanks})$
- **…**

Semantic Parsing with λ-DCS

(Berant et al., 2013)

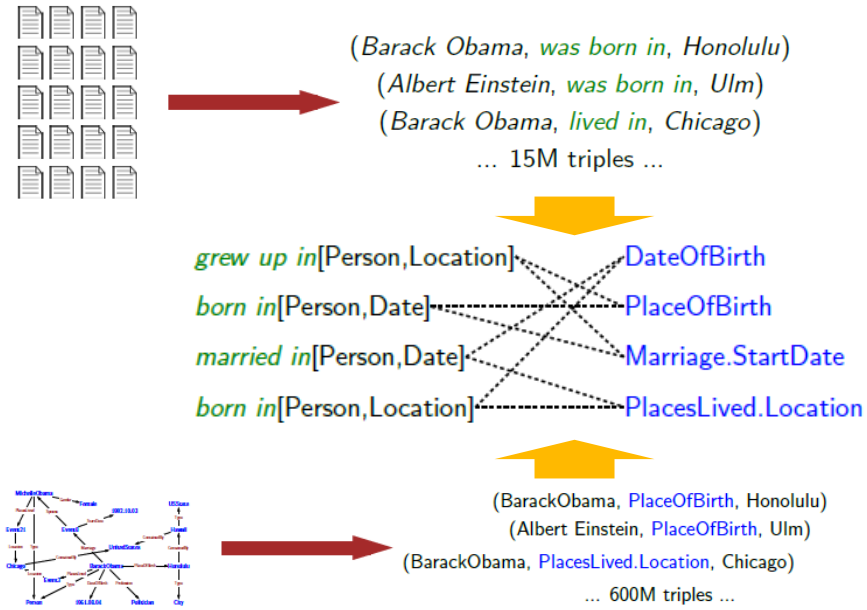
$$\lambda x.\,\mathrm{Type}(\mathrm{City}, x) \wedge \mathrm{Place\_of\_Birth}(\mathrm{Barack\ Obama}, x)$$

Type.City ⊓ Place_of_Birth.BarackObama



Alignment

- Map mentions to KB entities\predicates



Training Semantic Parser with Q-LF Pairs as Supervision

(Zettlemoyer and Collins, 2007)

$$\widehat{\mathrm{LF}} = \underset{\mathrm{LF}\in\{\mathrm{LF}_1,\ldots,\mathrm{LF}_n\}}{\operatorname{argmax}} \mathrm{P}(\mathrm{LF}|\mathrm{Q};\bar{\mathbf{w}}) = \underset{\mathrm{LF}\in\{\mathrm{LF}_1,\ldots,\mathrm{LF}_n\}}{\operatorname{argmax}} \sum_i \mathbf{w_i} \cdot \Phi_i(\mathrm{LF},\mathcal{Q})$$

But labeling logical forms for natural language is very expensive.

- Remember **Perception**?
- Update weight if there is a mistake

$$w_i^+ = w_i - \alpha\frac{\partial E}{\partial w_i} = w_i + \Phi_i(\mathrm{LF}^{\mathrm{true}},\mathcal{Q}) - \Phi_i(\widehat{\mathrm{LF}},\mathcal{Q})$$

  - increase score of positive examples
  - decrease score of negative examples
  - no change, if highest scoring answer is correct

Structured Perceptron Algorithm

```
create map w
for I iterations
    for each labeled pair X, Y_prime in the data
        Y_hat = HMM_VITERBI(w, X)
        phi_prime = CREATE_FEATURES(X, Y_prime)
        phi_hat = CREATE_FEATURES(X, Y_hat)
        w += phi_prime - phi_hat
```

Training Semantic Parser with QA Pairs as Weak Supervision

(Bao et al., 2014; Berant et al., 2013)

- Use answers as guides to train parameters in the semantic parser

$$\widehat{\mathrm{LF}} = \underset{\mathrm{LF}\in\{\mathrm{LF}_1,\ldots,\mathrm{LF}_n\}}{\operatorname{argmax}} \mathrm{P}(\mathrm{LF}|\mathrm{Q};\bar{\mathbf{w}}) = \underset{\mathrm{LF}\in\{\mathrm{LF}_1,\ldots,\mathrm{LF}_n\}}{\operatorname{argmax}} \sum_{\mathrm{i}} \mathbf{w_i} \cdot \Phi_{\mathrm{i}}(\mathrm{LF}, Q)$$

initial weights

$(1, 1, \ldots, 1)$

1^{st} round parsing

***Logical Form N*-best**		***Answer N*-best**	
T1	✗	A1	✗
T2	✗	A2	✗
T3	✓	A3	✓
T4	✗	A4	✗
T5	✓	A5	✓

Training Semantic Parser with QA Pairs as Weak Supervision

(Bao et al., 2014; Berant et al., 2013)

- Use answers as guides to train parameters in the semantic parser

$$\widehat{\mathrm{LF}} = \underset{\mathrm{LF}\in\{\mathrm{LF}_1,\ldots,\mathrm{LF}_n\}}{\operatorname{argmax}} \mathrm{P}(\mathrm{LF}|\mathrm{Q};\bar{\mathbf{w}}) = \underset{\mathrm{LF}\in\{\mathrm{LF}_1,\ldots,\mathrm{LF}_n\}}{\operatorname{argmax}} \sum_{\mathrm{i}} \mathbf{w_i} \cdot \Phi_{\mathrm{i}}(\mathrm{LF}, Q)$$

updated weights

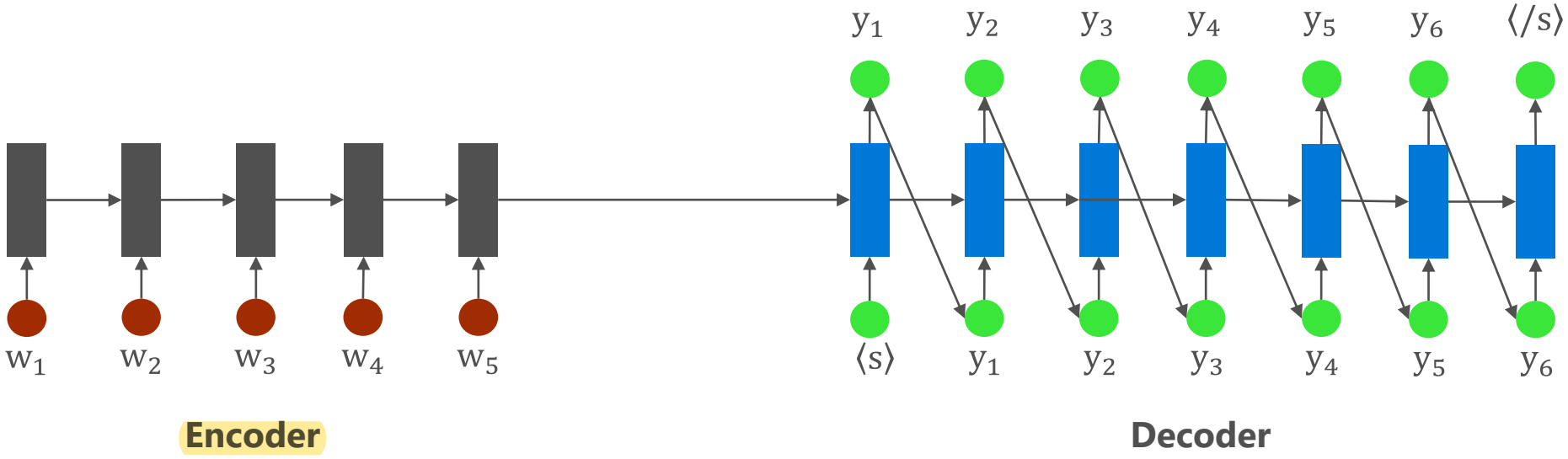
(0.2, -1.3, . . . , 0.7)

1st round optimization

***Logical Form N*-best**		***Answer N*-best**	
T3	✓	A3	✓
T5	✓	A5	✓
T1	✗	A1	✗
T4	✗	A4	✗
T2	✗	A2	✗

Generic Framework of Neural Network-based Semantic Parsing

- Perform semantic parsing as neural machine translation
  - **Encoder** considers a question as source language, encodes it into hidden states using RNN
  - **Decoder** considers a logical form as target language, generate it word-by-word based on question encoding using RNN

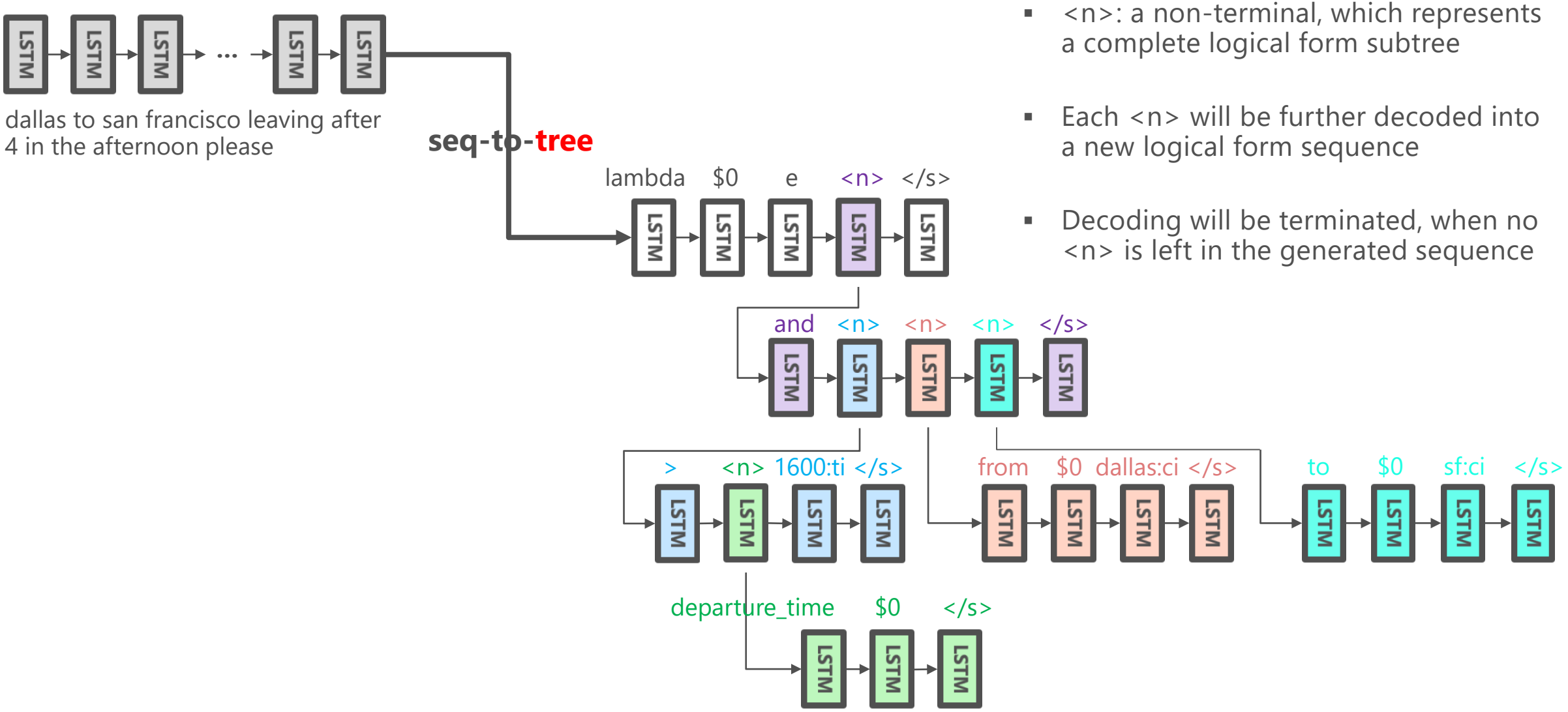


- Issue
  - Seq-to-Seq model ignores the **hierarchical structure** of logical forms

dallas to san francisco leaving after 4 in the afternoon please ➡ (Lambda $0 e (and (>(departure_time $0) 1600:ti) (from $0 dallas:ci) (to $0 sf:ci)))

Semantic Parsing with Seq-to-Tree Neural Network

(Dong and Lapata, 2016; Jia and Liang, 2016)



- <n>: a non-terminal, which represents a complete logical form subtree
- Each <n> will be further decoded into a new logical form sequence
- Decoding will be terminated, when no <n> is left in the generated sequence

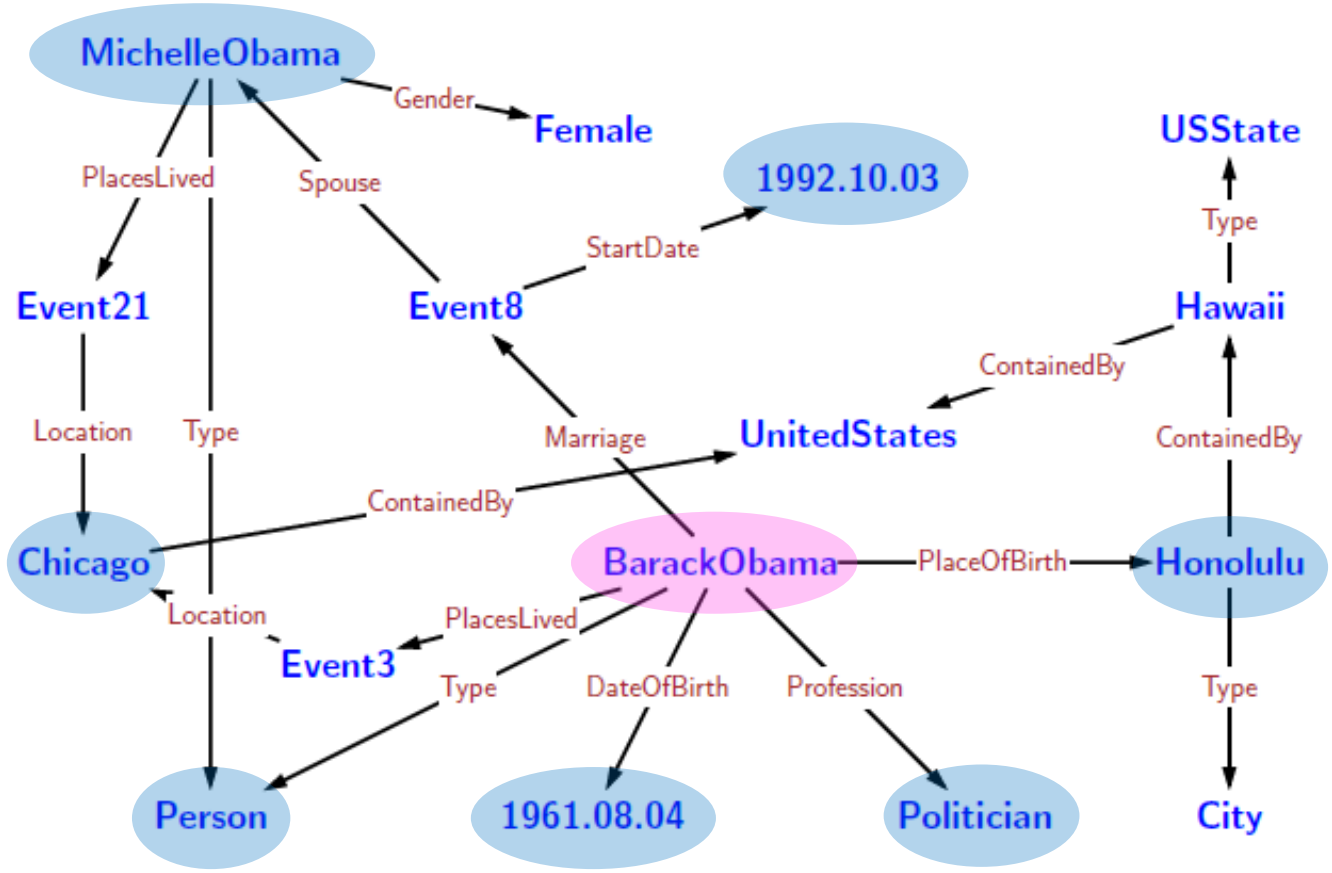
Answer Lookup

- Find answers by executing a logical form against KB is straightforward…

Outline of KBQA

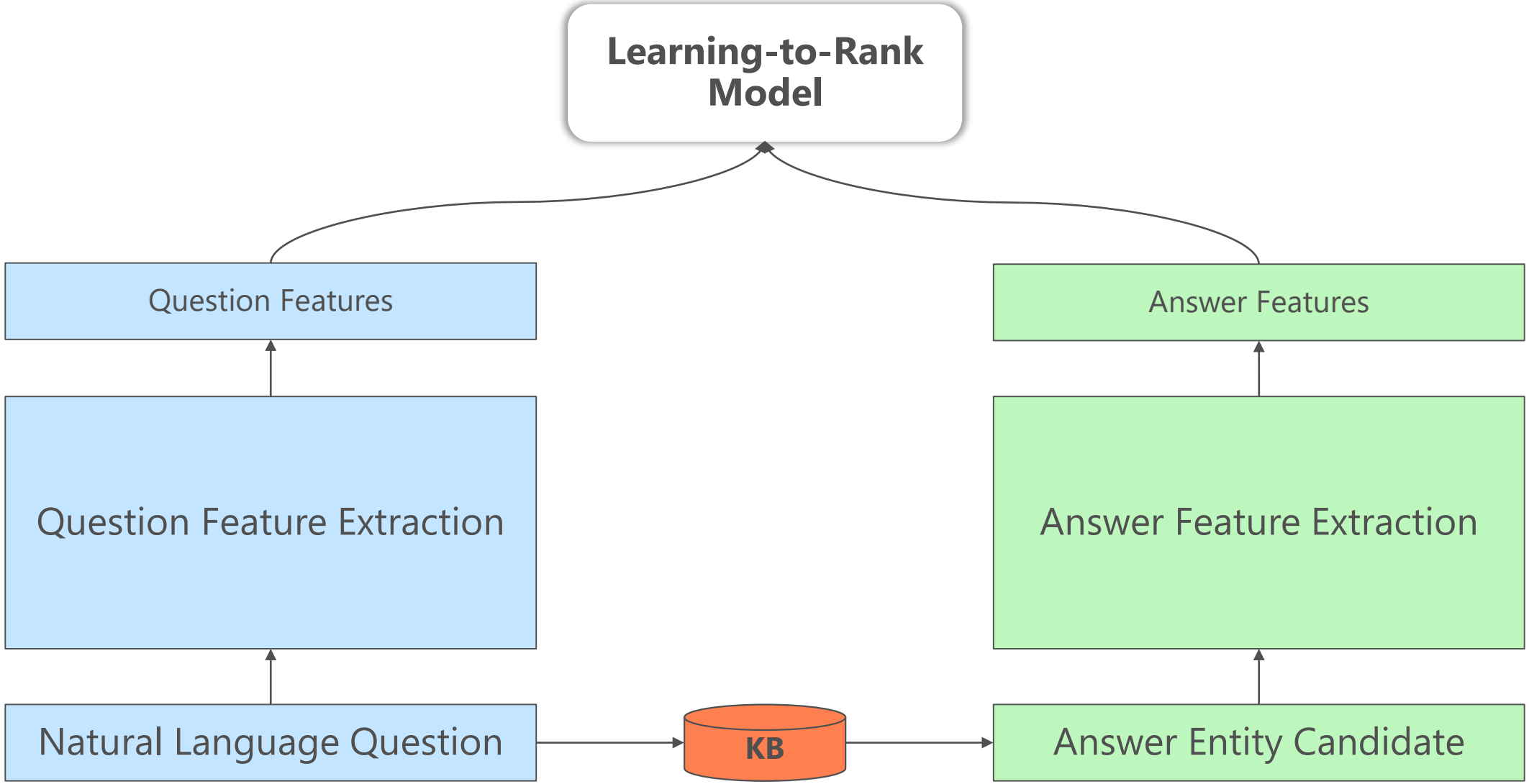
- Semantic Parsing-based KBQA
  1. What is **logical form**?
  2. How to **parse** a question into its logical form?
  3. How to **execute** a logical form against KB?
- Answer Ranking-based KBQA
  1. How to **select** answer candidates?
  2. How to **represent** answer candidates?
  3. How to **rank** answer candidates?

Answer Candidate Selection



- **Input question**
  - Where was Obama born?
- **Question entity detection**
  - Obama → Barack Obama
- **Answer candidate selection**
  - Entities connected to the question entity within n hops
  - Usually, $n = 1$ or 2

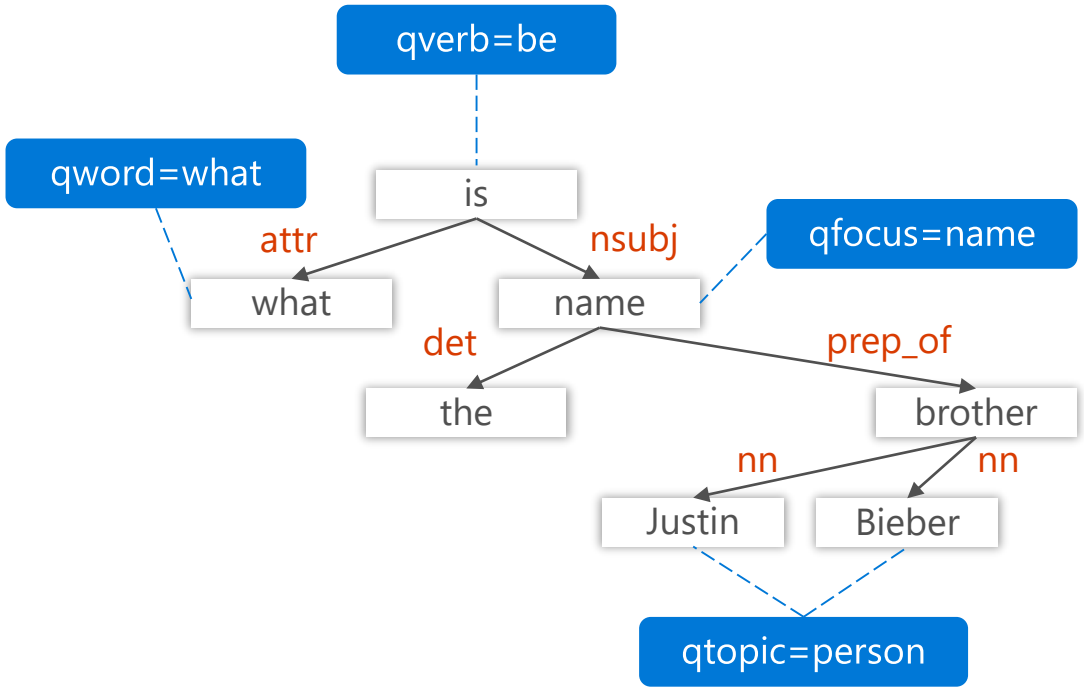
Answering Ranking with Features

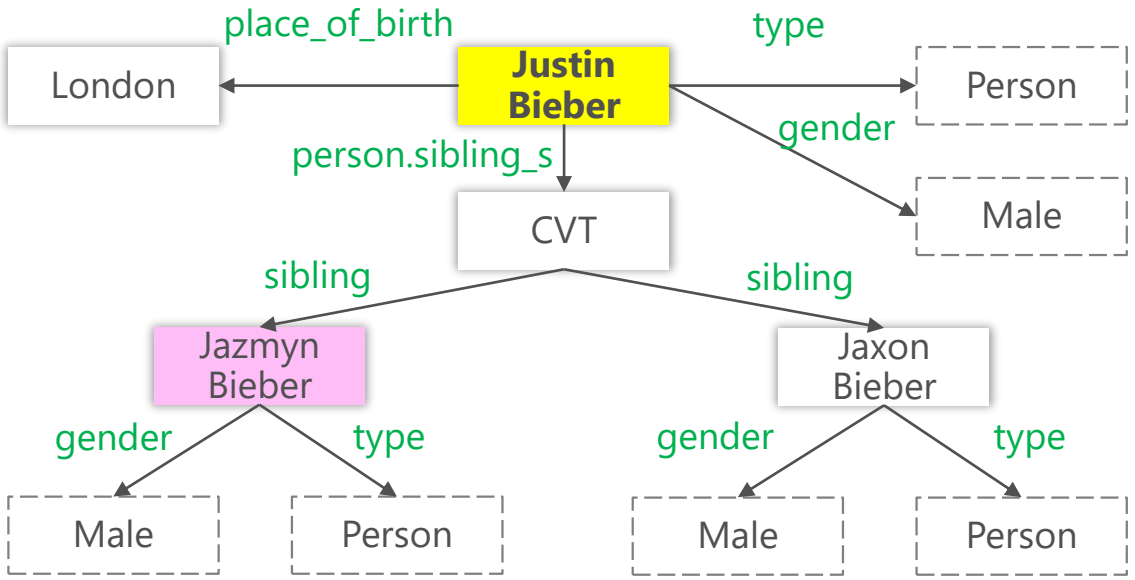


An Example

(Yao and Durme, 2014)

what is the name of Justin Bieber brother





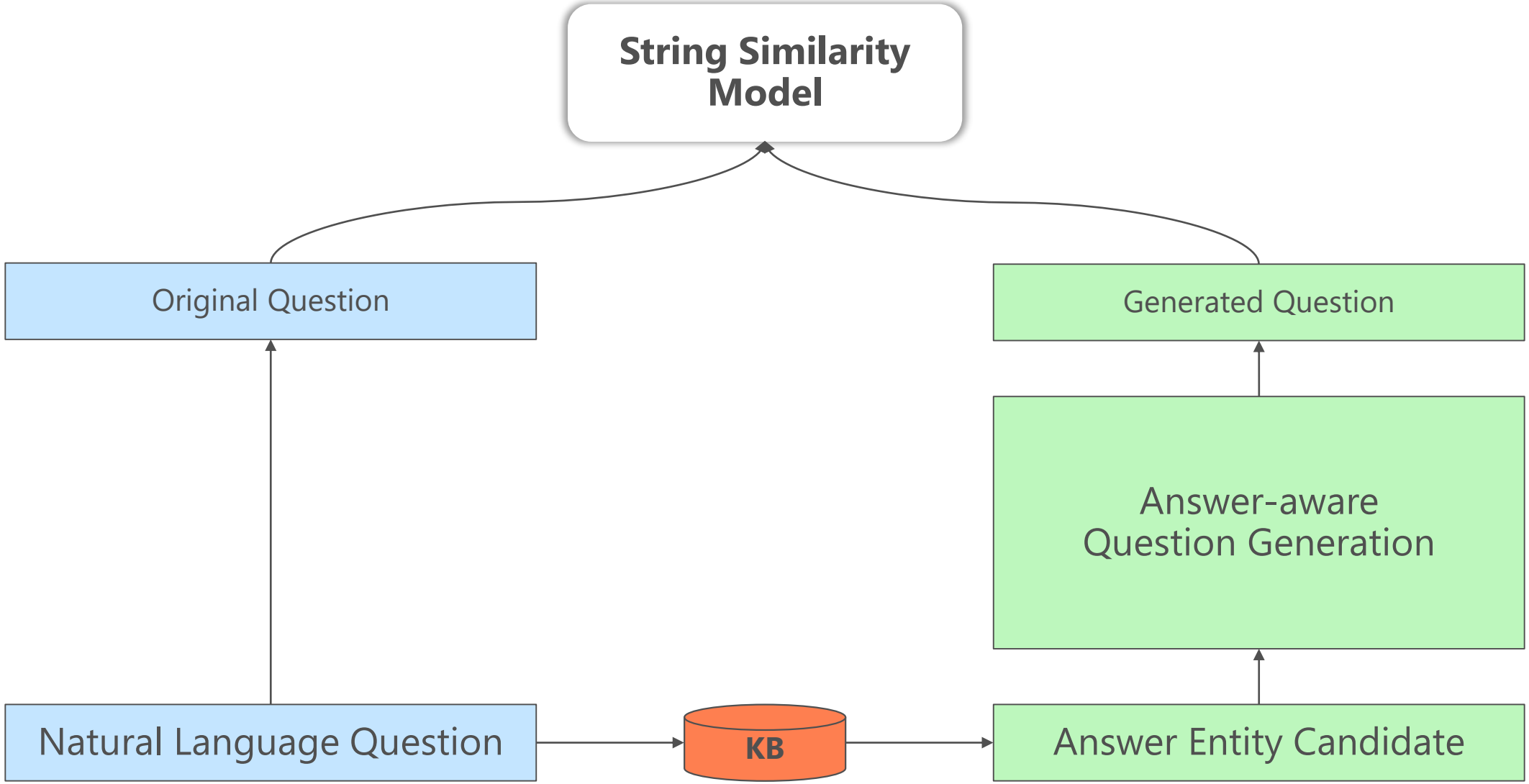
$$\tilde{A} = \underset{A}{\operatorname{argmax}} \sum_i \lambda_i \cdot h_i(QG, TG; A)$$

Each feature is a pairwise concatenation of a question graph feature and a topic graph feature of a specific answer node candidate.

Question Graph (QG)

Topic Graph (TG)

Answer Ranking with Question Generation



An Example

(Berant and Liang, 2014)

which city was Obama born ?

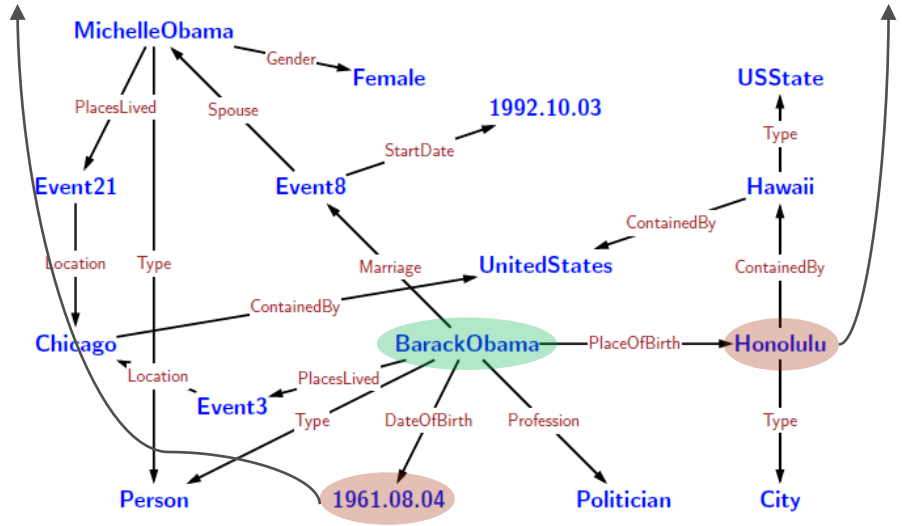
Ranking Model
(question-generated question)

When is the date of birth of Barack Obama ? ... What city is the place of birth of Barack Obama ?

Pattern-based Question Generation

DataOfBirth.BarackObama ... Type.City ∩ PlaceOfBirth.BarackObama

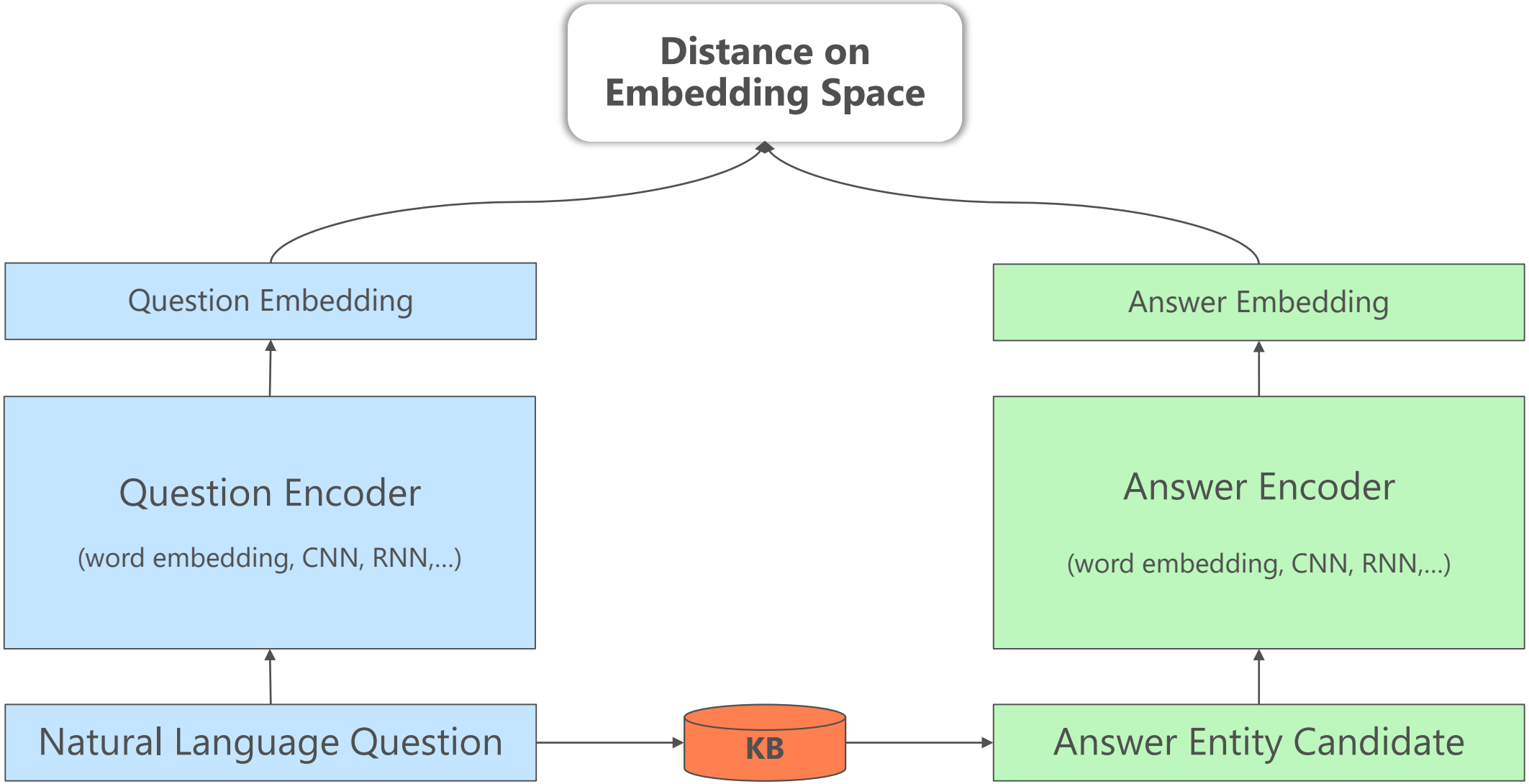
Predicate POS	Question Generation Pattern
NP	What **TYPE** is the **NP** of **ENTITY** ?
NP VP	What **NP** is **VP** by **ENTITY** ?
...	...



Ranking Model

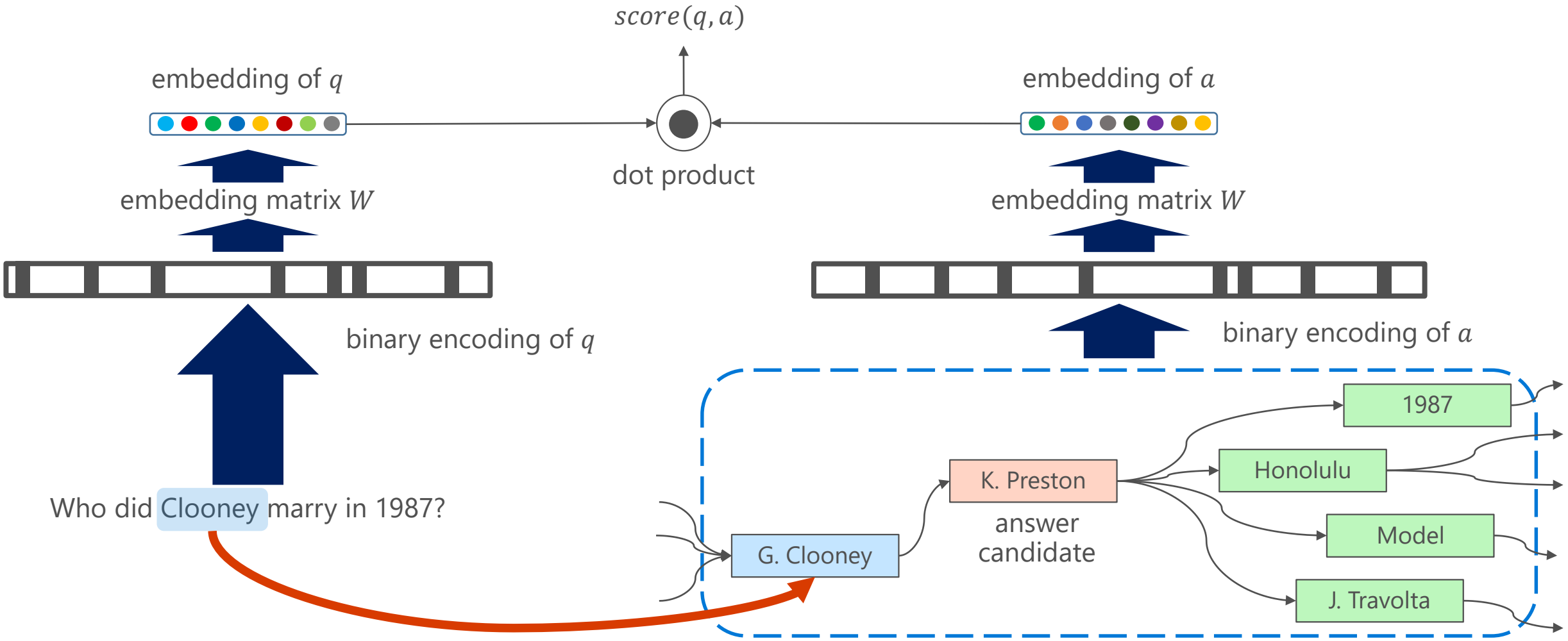
- Compute similarity between input question and each generated question
- Features
  - Association Model (paraphrasing)
  - Vector Space Model (word embedding)

Answering Ranking with Embedding

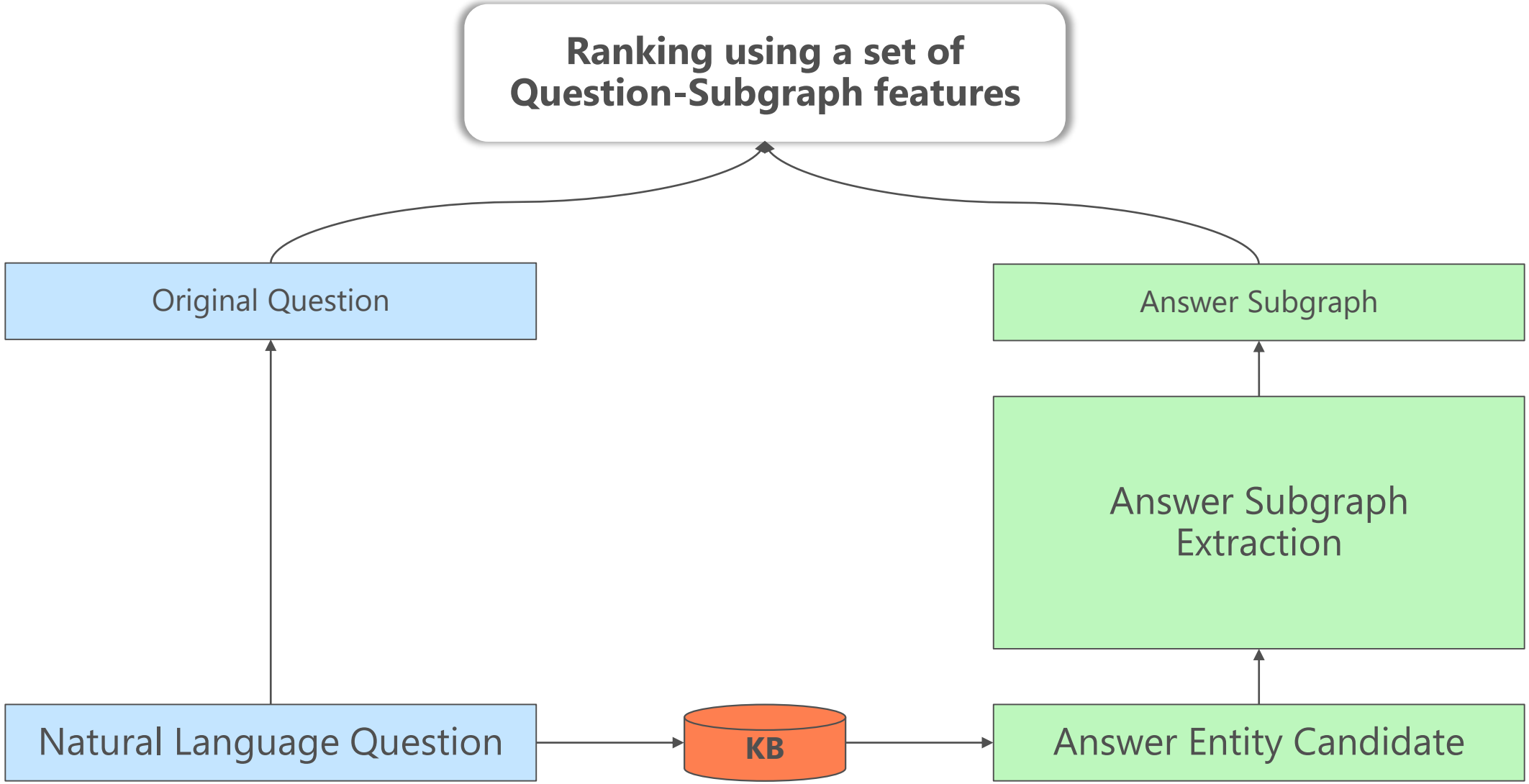


An Example

(Bordes et al., 2014)



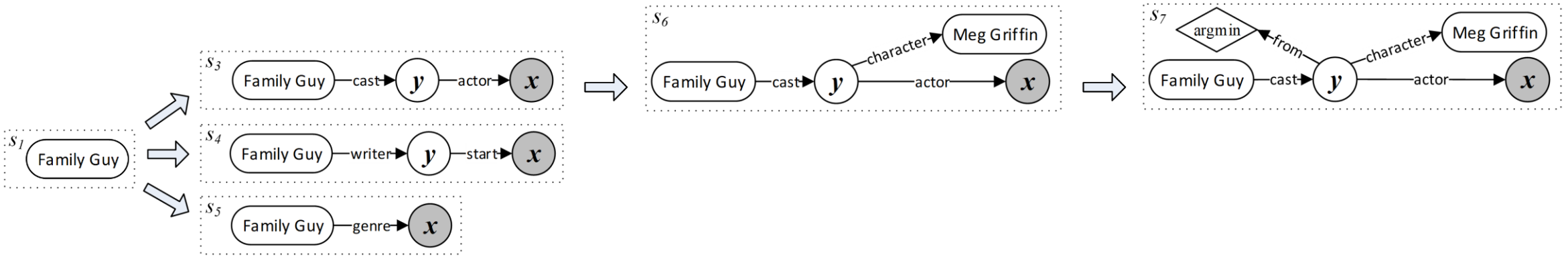
Answering Ranking with Subgraph

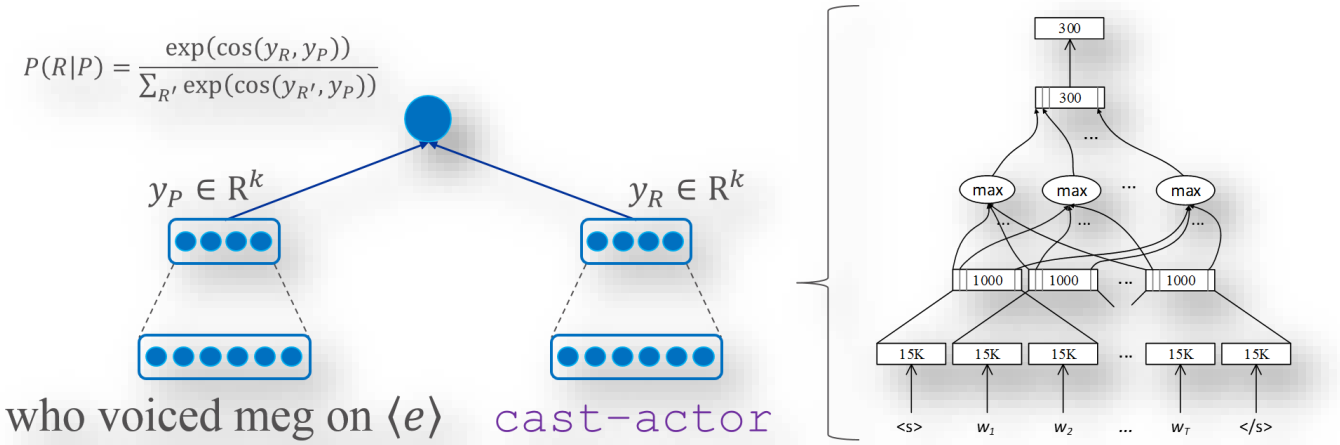


An Example

(Yih et al., 2015)

Who first voiced Meg on Family Guy ?





Answer Ranking with Memory Network

(Miller et al., 2016)

Each candidate memory is assigned a relevance probability by comparing its key with the question

$$p_{h_i} = \text{Softmax}(A\Phi_X(x) \cdot A\Phi_K(k_{h_i}))$$

H is a fixed size

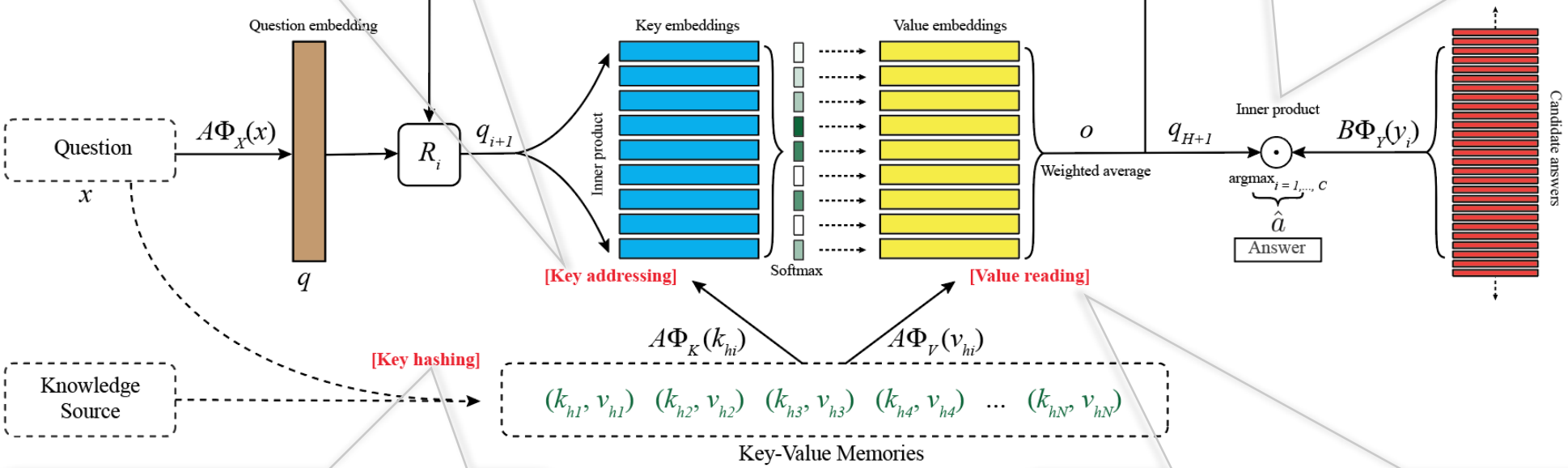
Hops $i = 1 \ldots H$

After receiving the result o, the query is updated with:

$$q_2 = R_1(q + o)$$

After a fixed number of H hops, the final prediction is:

$$\hat{a} = \text{argmax}_{i=1,\ldots,C}\text{Softmax}(q_{H+1}^{\top} B\Phi_Y(y_i))$$



Finds a subset of K-V pairs, where each key shares at least one word with the question

K-V pairs

- <sbj+pred, obj>
- <sentence, sentence>
- <window, center word>
- …

The values of the memories are read by taking their weighted sum using the addressing probabilities, and o is returned.

$$o = \sum_i p_{h_i} A\Phi_V(v_{h_i})$$

Semantic Parsing-based KBQA vs. Answer Ranking-based KBQA

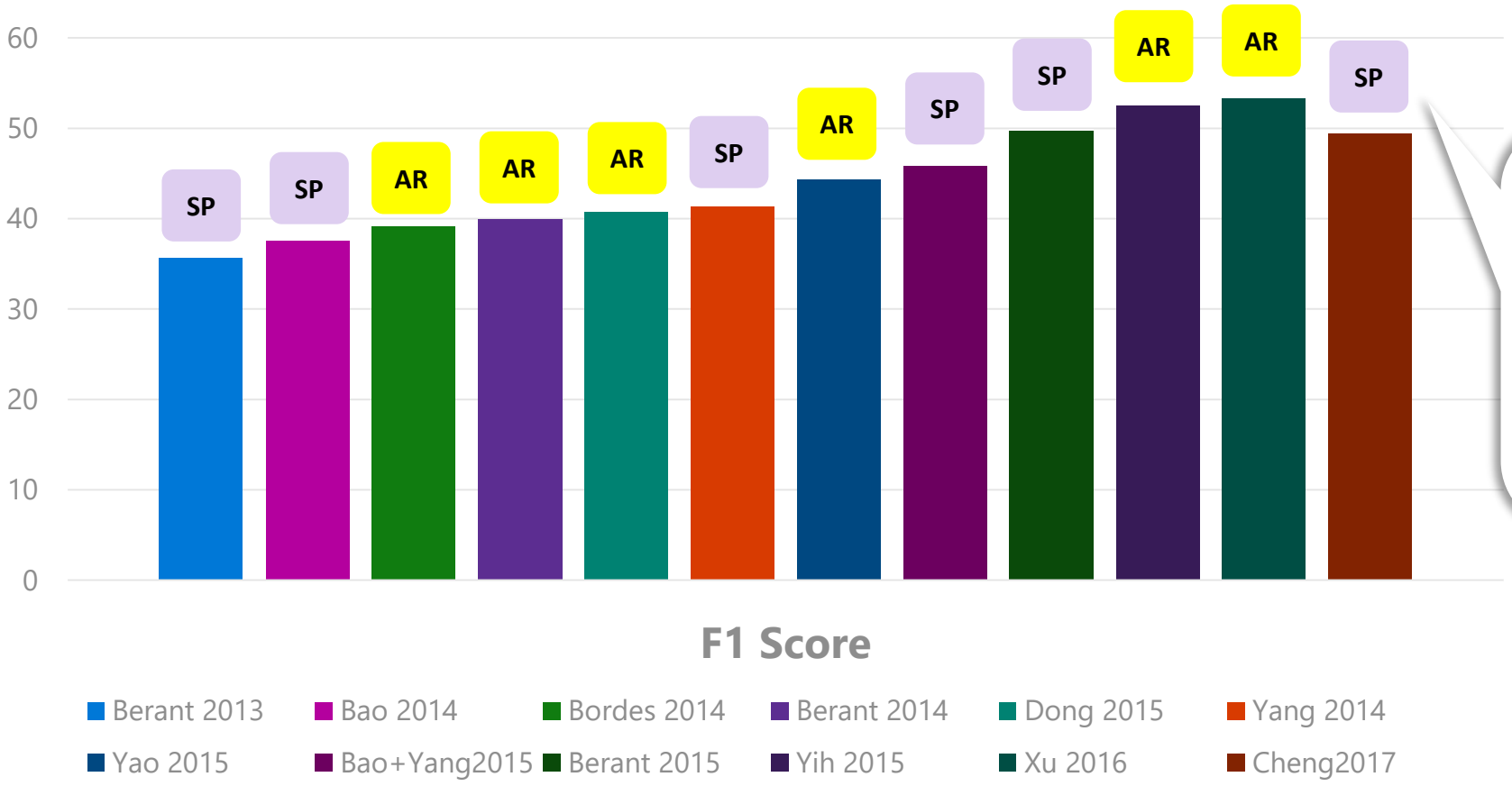
- Compared on **WebQuestions dataset**
  - http://nlp.stanford.edu/software/sempre/
- Data statistic
  - 5,810 Q-A pairs (English) (questions are sampled from Google query log)
  - Most of them are one-hop factoid questions
- Citation
  - Jonathan Berant, Andrew Chou, Roy Frostig, Percy Liang. Semantic Parsing on Freebase from QA Pairs. EMNLP, 2013
- Data example

  what is the name of Justin Bieber brother ?
  http://www.freebase.com/view/en/justin_bieber
  Jazmyn Bieber

  what state does Salena Gomez ?
  http://www.freebase.com/view/en/selena_gomez
  New York City

Semantic Parsing-based KBQA vs. Answer Ranking-based KBQA

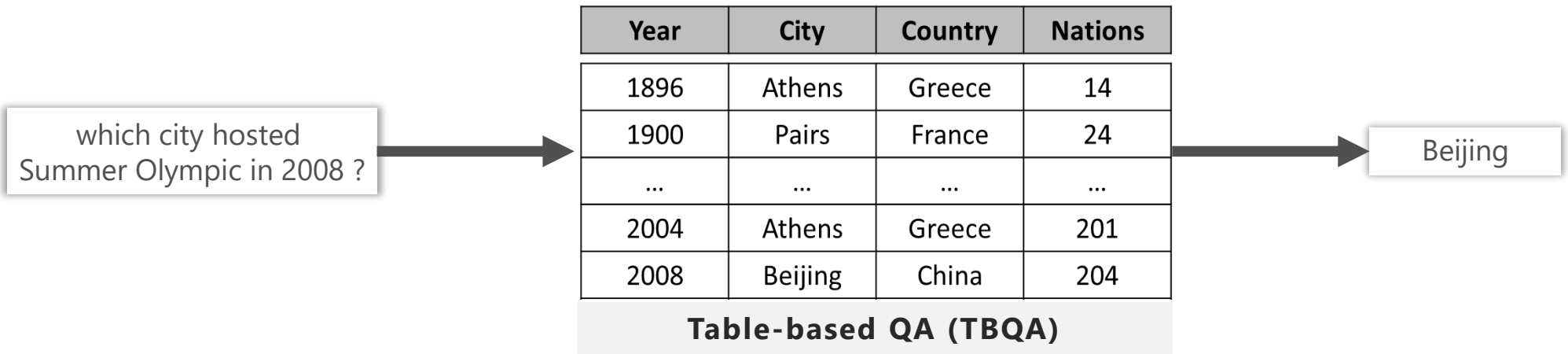
- Evaluation results on WebQuestions using F1 Score as metric



- **AR**: Answer Ranking-based KBQA
- **SP**: Semantic Parsing-based KBQA
- AR-based methods perform better than SP-based methods

Table-based QA (TBQA)

- Two types of approaches
  - Semantic parsing-based
  - Answer ranking-based



Table

- Semi-structured data with flexible schema

Year	City	Country	Nations
1896	Athens	Greece	14
1900	Pairs	France	24
...	...	...	...
2004	Athens	Greece	201
2008	Beijing	China	204

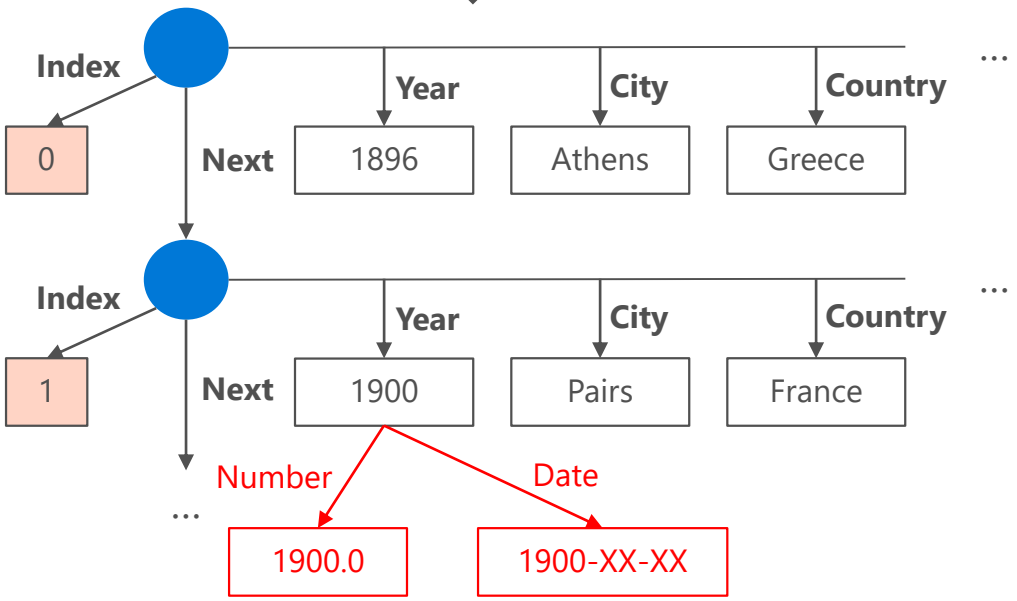
Host Cities of Summer Olympic Games

- **Table Cell**
  Objects/Values in the world
- **Table Header**
  The type of table cells in the same column
- **Table Row**
  A record of an information piece
- **Table Caption**
  Summary of the entire table

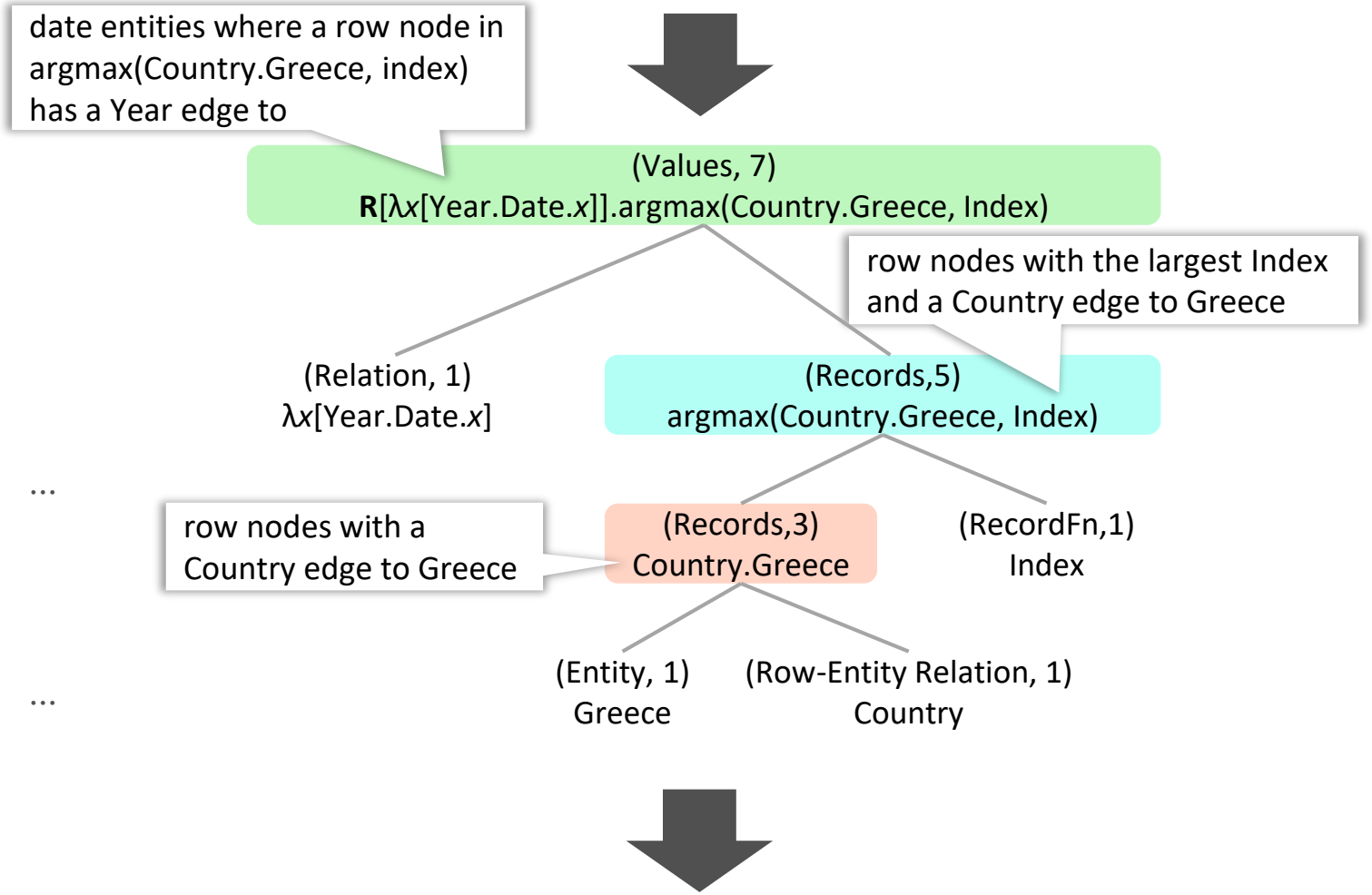
Semantic Parsing-based TBQA with Deduction Rules

(Pasupat and Liang, 2015)

Year	City	Country	Nations
1896	Athens	Greece	14
1900	Pairs	France	24
…	…	…	…
2004	Athens	Greece	201
2008	Beijing	China	204



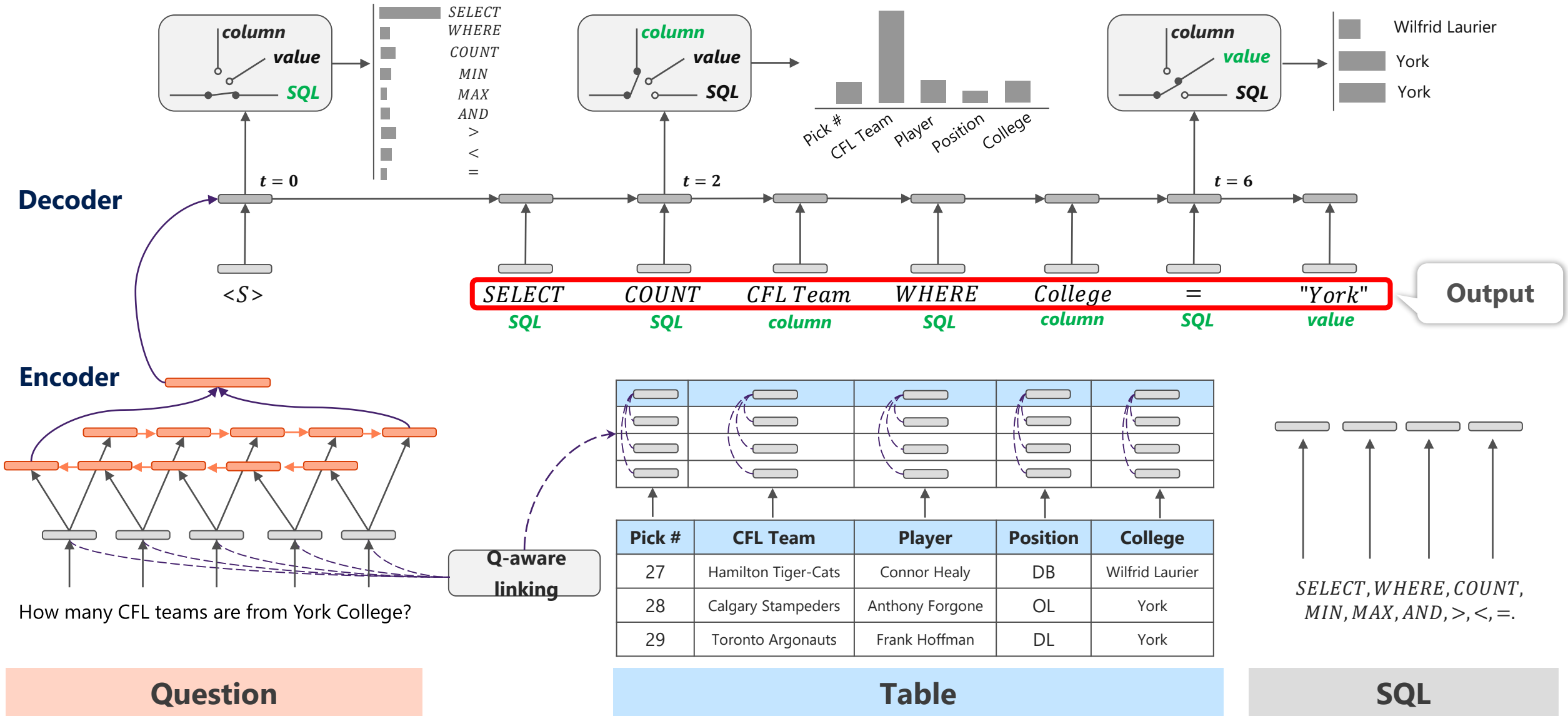
Question: when did Greece hold its last Summer Olympics?



Answer: 2004

Semantic Parsing-based TBQA with Seq-to-SQL

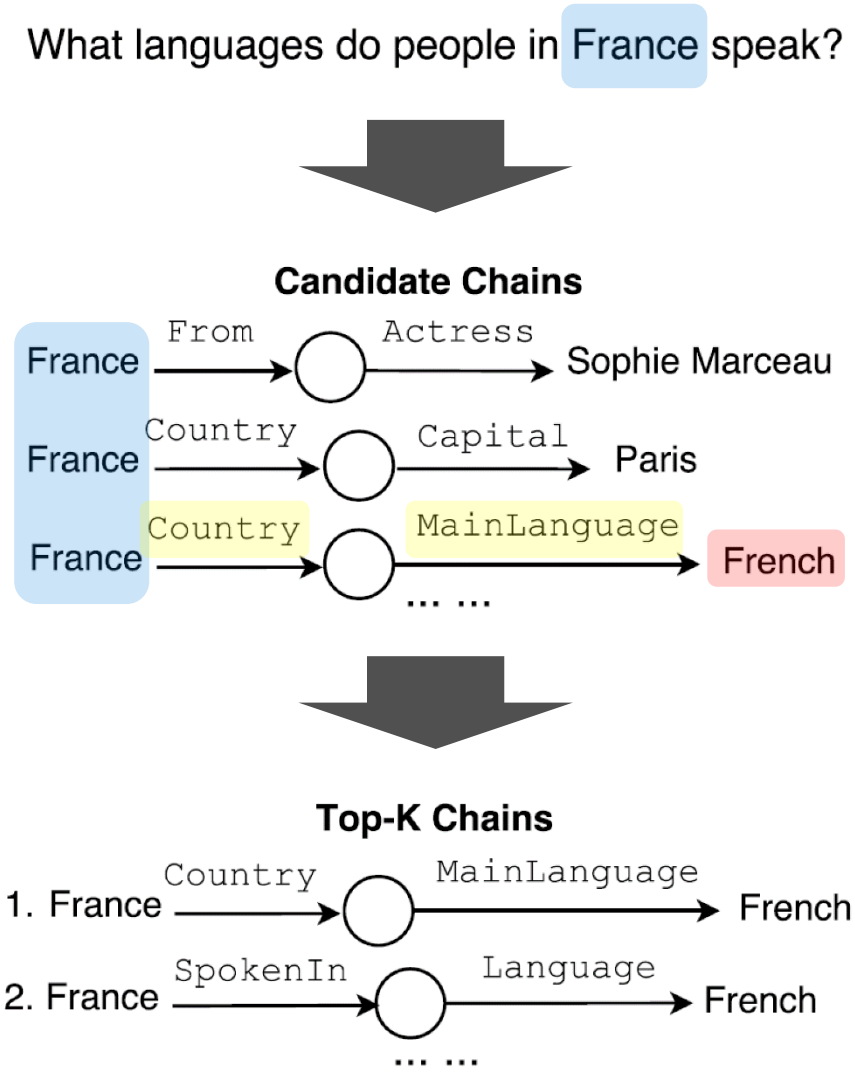
(Sun et al., 2018)



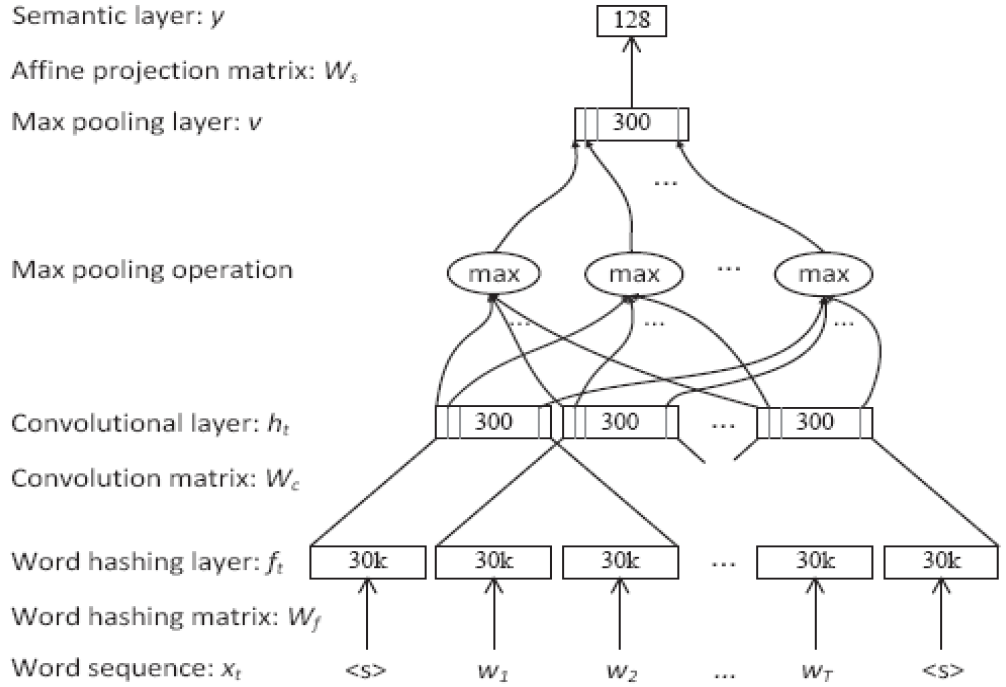
Pick #	CFL Team	Player	Position	College
27	Hamilton Tiger-Cats	Connor Healy	DB	Wilfrid Laurier
28	Calgary Stampeders	Anthony Forgone	OL	York
29	Toronto Argonauts	Frank Hoffman	DL	York

Answer Ranking-based TBQA with Features

(Sun et al., 2016; Jauhar et al., 2016)

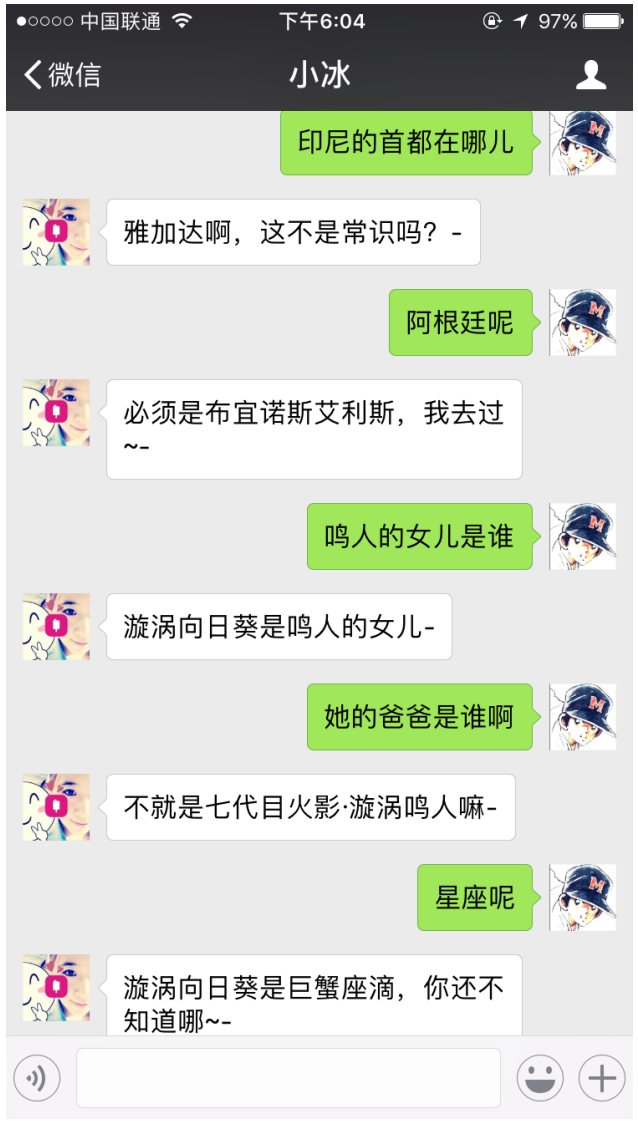


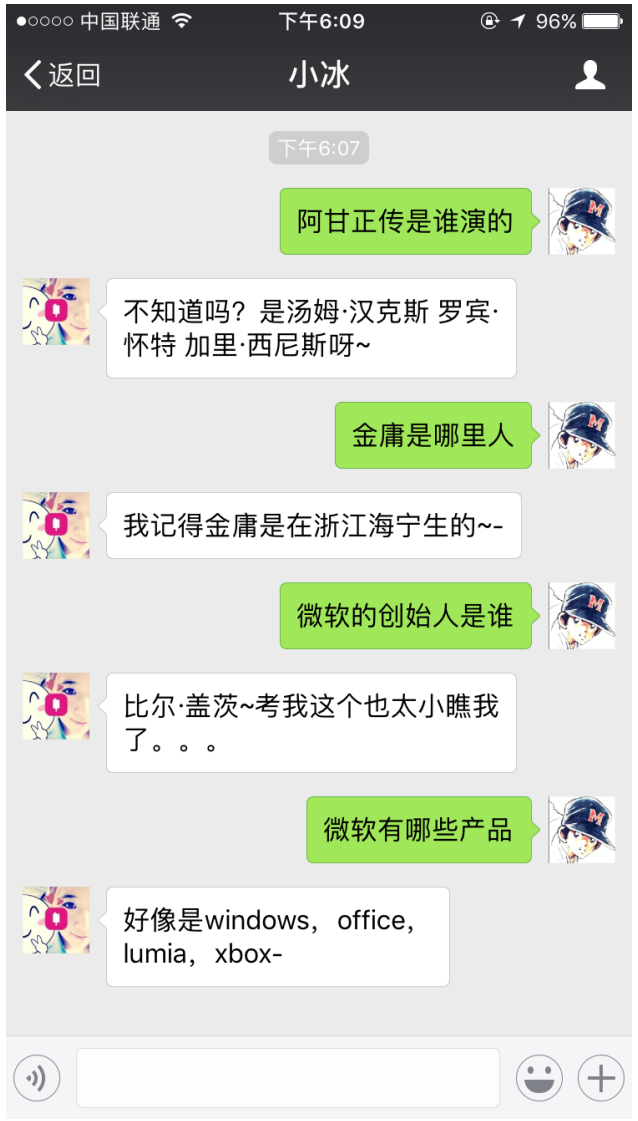
Country	Capital	Currency	Main Language
Algeria	Algiers	Dinar	Arabic
Egypt	Cairo	Pound	Arabic
France	Paris	Euro	French
...	...	...	...



KBQA Applications

KBQA in Chat

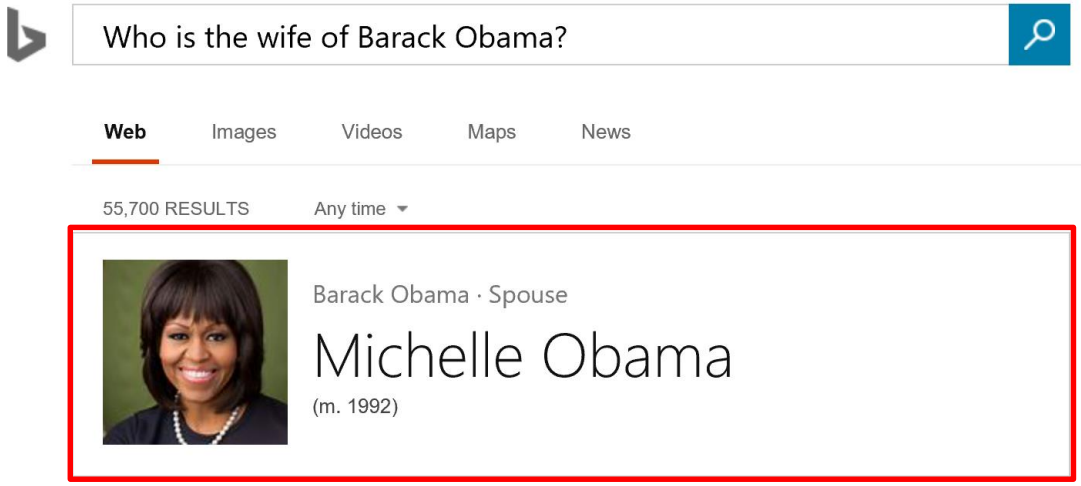


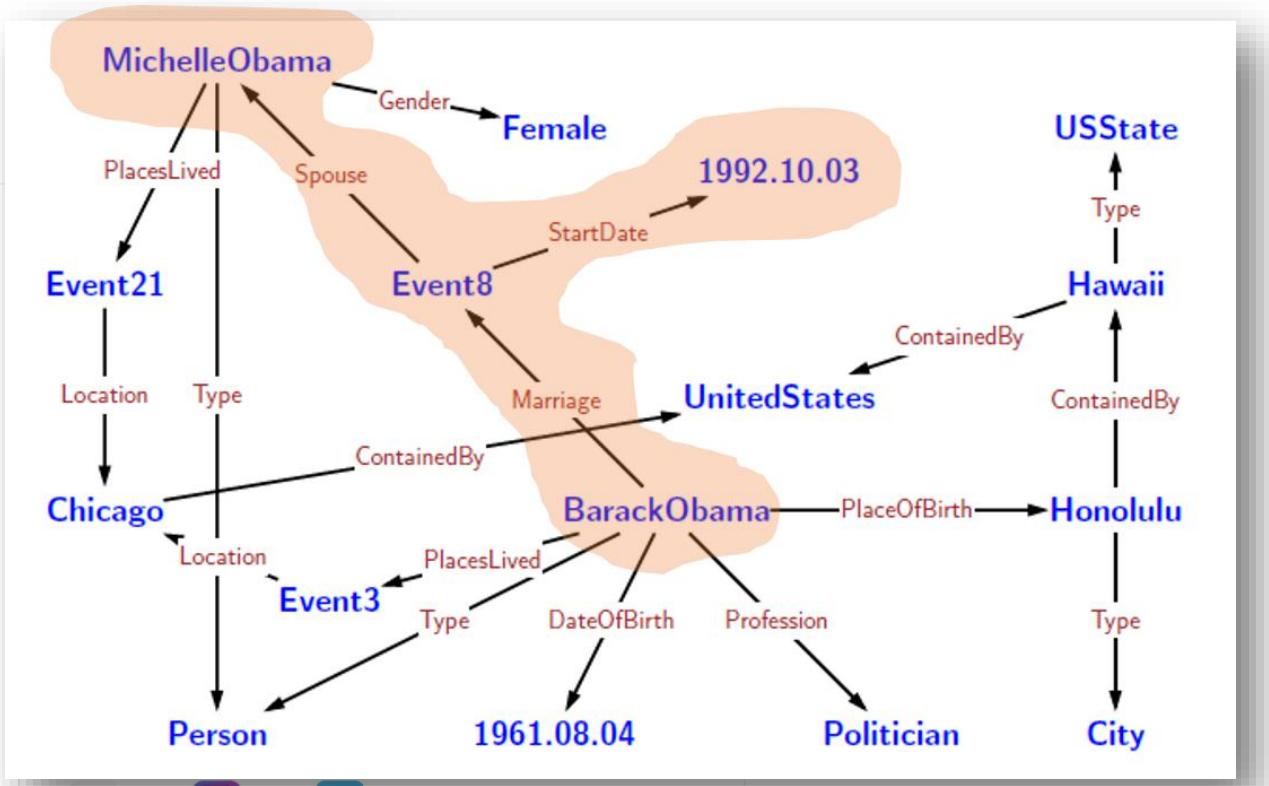


Chinese Knowledge Graph (BingKnows)

```
微软 ||| 公司名称 ||| 微软公司↓
微软 ||| 外文名称 ||| Microsoft corporation.↓
微软 ||| 总部地点 ||| 美国华盛顿州雷德蒙市↓
微软 ||| 成立时间 ||| 1975年4月4日16时↓
微软 ||| 经营范围 ||| 操作系统，办公软件，手机↓
微软 ||| 公司性质 ||| 上市公司、外商独资↓
微软 ||| 公司口号 ||| 新效率(New Efficiency)↓
微软 ||| 年营业额 ||| 77,849百万美元（2014年）↓
微软 ||| 员工数 ||| 99,000人(2014年)↓
微软 ||| 联合创始人 ||| 比尔·盖茨、保罗·艾伦↓
微软 ||| 现任董事长 ||| 约翰·汤普森↓
微软 ||| 首席执行官 ||| 萨蒂亚·纳德拉↓
微软 ||| 首席运营官 ||| 凯文·特纳↓
微软 ||| 世界500强 ||| 第104位（2014年）↓
微软 ||| 成立地点 ||| 美国新墨西哥州阿尔伯克基市↓
微软 ||| 中国总部 ||| 中国北京海淀区知春路49号↓
微软 ||| 主要产品 ||| xbox，windows，office，lumia↓
董明珠(珠海格力集团有限公司原董事长) ||| 中文名 ||| 董明珠↓
董明珠(珠海格力集团有限公司原董事长) ||| 外文名 ||| Mingzhu Dong↓
董明珠(珠海格力集团有限公司原董事长) ||| 别名 ||| 东方明珠↓
董明珠(珠海格力集团有限公司原董事长) ||| 国籍 ||| 中华人民共和国↓
董明珠(珠海格力集团有限公司原董事长) ||| 民族 ||| 汉↓
董明珠(珠海格力集团有限公司原董事长) ||| 出生地 ||| 江苏南京↓
董明珠(珠海格力集团有限公司原董事长) ||| 出生日期 ||| 1954年8月↓
董明珠(珠海格力集团有限公司原董事长) ||| 职业 ||| 格力电器董事长兼总裁
董明珠(珠海格力集团有限公司原董事长) ||| 毕业院校 ||| 芜湖职业技术学院
董明珠(珠海格力集团有限公司原董事长) ||| 主要成就 ||| 全球100位最佳CEO
董明珠(珠海格力集团有限公司原董事长) ||| 代表作品 ||| 《棋行天下》↓
```

KBQA in Search





Wikipedia Instagram Twitter

Spouse: Barack Obama (m. 1992)

Born: Jan 17, 1964 (age 53) · Chicago, IL

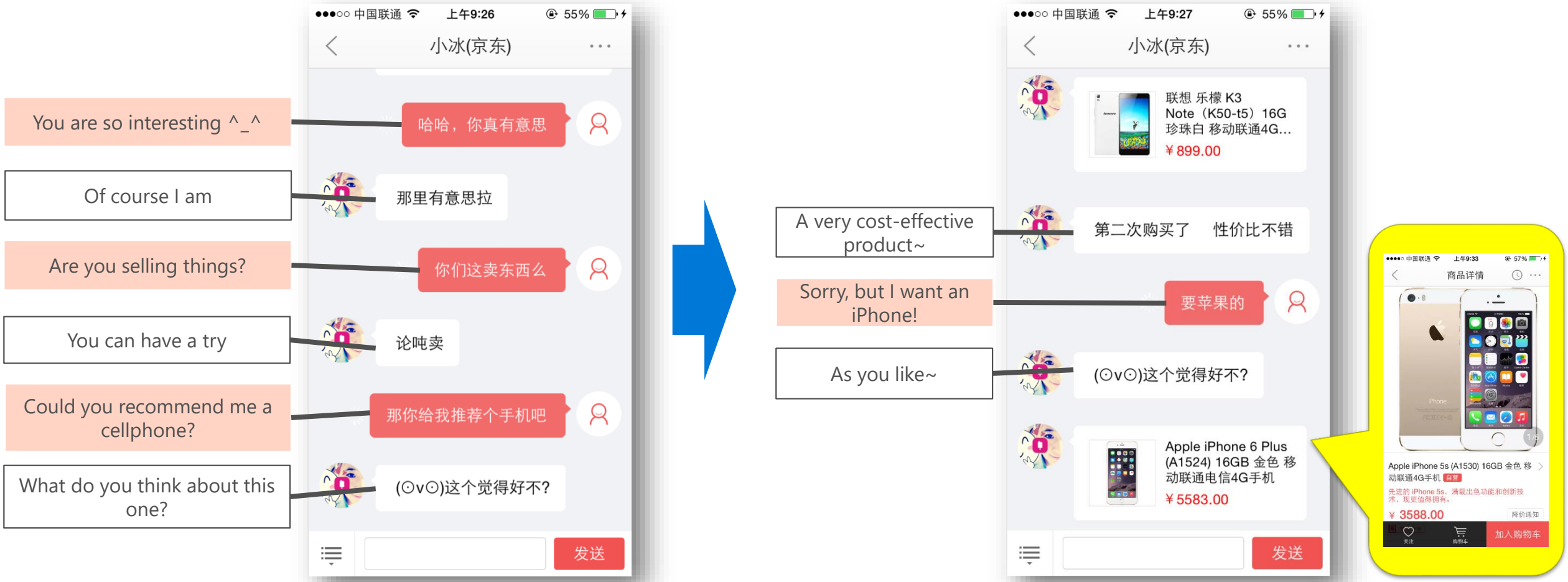
Height: 5' 11" (1.80 m)

Children: Sasha Obama (Daughter) · Malia Obama (Daughter)

Education: Princeton University · Harvard Law School (1985 - 1988) · Young Magnet High School

Parents: Marian Shields Robinson (Mother) · Fraser C. Robinson III (Father)

KBQA in Task Completion



KBQA Datasets

KBQA Datasets

- KBQA (English)
  - **WebQuestions** (Stanford)
    - https://nlp.stanford.edu/software/sempre/
  - **SimpleQuestions** (Facebook)
    - https://research.fb.com/downloads/babi/
  - **LC-QuAD** (University of Bonn)
    - http://lc-quad.sda.tech/
  - **ComplexWebQuestions** (Tel-Aviv University)
    - https://www.tau-nlp.org/compwebq
  - **Complex Sequential Question Answering** (IBM)
    - https://amritasaha1812.github.io/CSQA/
- TableQA (English)
  - **WikiTableQuestions** (Stanford)
    - https://nlp.stanford.edu/blog/wikitablequestions-a-complex-real-world-question-understanding-dataset/
  - **WikiSQL** (Salesforce)
    - https://github.com/salesforce/WikiSQL
- KBQA (Chinese)
  - **NLPCC2016-KBQA** (NLPCC)
    - http://tcci.ccf.org.cn/conference/2016/pages/page05_evadata.html
  - **NLPCC2017-KBQA** (NLPCC)
    - http://tcci.ccf.org.cn/conference/2017/taskdata.php
  - **NLPCC2018-KBQA** (NLPCC)
    - http://tcci.ccf.org.cn/conference/2018/taskdata.php

The SimpleQuestions dataset

This section proposes SimpleQuestions, a dataset collected for research in automatic question answering with human generated questions. Details and baseline results on this dataset can be found in the paper:

Antoine Bordes, Nicolas Usunier, Sumit Chopra and Jason Weston. Large-Scale Simple Question answering with Memory Networks, arXiv:1506.02075.

The SimpleQuestions dataset consists of a total of 108,442 questions written in natural language by human English-speaking annotators each paired with a corresponding fact, formatted as (subject, relationship, object), that provides the answer but also a complete explanation. Facts have been extracted from the Knowledge Base Freebase. We randomly shuffle these questions and use 70% of them (75910) as training set, 10% as validation set (10845), and the remaining 20% as test set.

Here are some examples of questions and facts:

```
* What American cartoonist is the creator of Andy Lippincott?
  Fact: (andy_lippincott, character_created_by, garry_trudeau)
* Which forest is Fires Creek in?
  Fact: (fires_creek, containedby, nantahala_national_forest)
* What does Jimmy Neutron do?
  Fact: (jimmy_neutron, fictional_character_occupation, inventor)
* What dietary restriction is incompatible with kimchi?
  Fact: (kimchi, incompatible_with_dietary_restrictions, veganism)
```

LC-QuAD

Largescale Complex Question Answering Dataset

Download

OR

See Examples

Data Characteristics

Current Version	1.0
Total Questions	5000
Unique Templates	38
Entities Covered	5042
Predicates Covered	615

Contact Us

In case you find any bug in our framework, or any issue with our dataset, please inform us on Issues Page.

Contact privansh.trivedi@uni-

What is LC-QuAD?

The aim of LC-QuAD is to make a large dataset for Question Answering (QA) over structured data (in RDF format) available. It consists of 5000 pairs of natural language question and the corresponding SPARQL query. In order to create the dataset, we used a set of typical query templates and then converted seed entities in the RDF graph to a normalised natural natural question structure (NNQS). This was then transformed to natural language questions with different lexical and syntactical variations by English native speakers. Please see our paper for more details.

Examples

```
Q: What are the mascots of the teams participating in
        the turkish handball super league?

SELECT DISTINCT ?uri WHERE {
  ?x dbp:league dbr:Turkish_Handball_Super_League .
  ?x dbp:mascot ?uri
}
```

Documentation & Usage Guides

A dataset for answering complex questions that require reasoning over multiple web snippets.

ComplexWebQuestions is a new dataset that contains a large set of complex questions in natural language, and can be used in multiple ways:

1. By interacting with a search engine, which is the focus of our paper (Talmor and Berant, 2018);
2. As a reading comprehension task: we release **9,595,163 web snippets** that are relevant for the questions, and were collected during the development of our model;
3. As a semantic parsing task: each question is paired with a SPARQL query that can be executed against Freebase to retrieve the answer.

Question: The actress that had the role of Martha Alston, plays what role in Finding Nemo?
Answer: "Dory"

Title: "Ellen DeGeneres - Wikipedia"
Web Snippet: "..She also played Martha Alston in the 1996 Touchstone Pictures film Mr. Wrong and.. "

Title: "Ellen DeGeneres | Disney Wiki | FANDOM powered by Wikia"
Web Snippet: "... provided the voice of Dory in Disney/Pixar's 2003 animated film, Finding Nemo. ..."

The dataset contains 34,689 examples, each containing:

- A complex question
- Answers (including aliases)
- An average of 276.6 snippets per question
- A SPARQL query (against Freebase)

Sample Questions

- “Which school that Sir Ernest Rutherford attended has the latest founding date?”
- “what movies does Leo Howard play in and that is 113.0 minutes long?"
- “Where is the end of the river that originates in Shannon Pot?”

Complex Sequential Question Answering: Towards Learning to Converse Over Linked Question Answer Pairs with a Knowledge Graph

Abstract

While conversing with chatbots, humans typically tend to ask many questions, a significant portion of which can be answered by referring to large-scale knowledge graphs (KG). While Question Answering (QA) and dialog systems have been studied independently, there is a need to study them closely to evaluate such real-world scenarios faced by bots involving both these tasks. Towards this end, we introduce the task of Complex Sequential QA which combines the two tasks of (i) answering factual questions through complex inferencing over a realistic-sized KG of millions of entities, and (ii) learning to converse through a series of coherently linked QA pairs. Through a labor intensive semi-automatic process, involving in-house and crowdsourced workers, we created a dataset containing around 200K dialogs with a total of 1.6M turns. Further, unlike existing large scale QA datasets which contain simple questions that can be answered from a single tuple, the questions in our dialogs require a larger subgraph of the KG. Specifically, our dataset has questions which require logical, quantitative, and comparative reasoning as well as their combinations. This calls for models which can: (i) parse complex natural language questions, (ii) use conversation context to resolve coreferences and ellipsis in utterances, (iii) ask for clarifications for ambiguous queries, and finally (iv) retrieve relevant subgraphs of the KG to answer such questions. However, our experiments with a combination of state of the art dialog and QA models show that they clearly do not achieve the above objectives and are inadequate for dealing with such complex real world settings. We believe that this new dataset coupled with the limitations of existing models as reported in this paper should encourage further research in Complex Sequential QA.

CODE

To be made available soon!

PAPER

Please download the paper here paper link

AAAI 2018 SLIDES

Please download the slides here slides link

BIBTEX

```
@article{1801.10314,
Author = {Amrita Saha and Vardaan Pahuja and Mitesh M. Khapra and Karthik Sankaranarayanan and Sarath Chandar},
Title = {Complex Sequential Question Answering: Towards Learning to Converse Over Linked Question Answer Pairs with a Knowledge Graph},
Year = {2018},
Eprint = {arXiv:1801.10314},
}
```

DATASET

Please click here to download the dataset.
NEW: We have revised the dialogs after incorporating some more feedback from users. (**DATED March 29, 2018**).

~~***NEW***~~: Some slight renaming of JSON fields done in the dialog zip. (**DATED March 15, 2018**).

~~***NEW***~~: We have revised the dialog and wikidata jsons after incorporating feedback from several users. All users are requested to re-download the entire data inclusive of wikidata and dialog JSONs. (**DATED March 6, 2018**).

Chinese KBQA Task in NLPCC

- **K**nowledge-**B**ased **Q**uestion **A**nswering (**KBQA**) task

- An example
  - <question id=28>　　新版还珠格格的导演是谁
  - <answer id=28>　　李平，丁仰国

	# of <Question, Triple, Answer> Triples
Train set	14,609
Test set	9,870

	# of <Subject, Predicate, Object> Triples
Chinese KB	47,943,429

```
新还珠格格 ||| entity.primaryName ||| 新还珠格格
新还珠格格 ||| 中文名 ||| 新还珠格格
新还珠格格 ||| 外文名 ||| New my fair Princess
新还珠格格 ||| 出品时间 ||| 2011年和2014年
新还珠格格 ||| 出品公司 ||| 上海创翊文化传播有限公司
新还珠格格 ||| 制片地区 ||| 中国大陆，中国台湾
新还珠格格 ||| 拍摄地点 ||| 横店影视城
新还珠格格 ||| 发行公司 ||| 上海创翊文化传播有限公司
新还珠格格 ||| 首播时间 ||| 2011年7月16日
新还珠格格 ||| 导演 ||| 李平，丁仰国
新还珠格格 ||| 编剧 ||| 琼瑶，黄素媛
新还珠格格 ||| 主演 ||| 李晟，海陆，张睿，李佳航，潘杰明，赵丽颖，邱心志，邓萃雯，刘雪华
新还珠格格 ||| 集数 ||| 总共98集→第一部1至37集→第二部37至74集→第三部74至98集
新还珠格格 ||| 每集长度 ||| 前三部：45分钟 第四部：48分钟
新还珠格格 ||| 类型 ||| 古装，爱情，励志，喜剧
新还珠格格 ||| 上映时间 ||| 前三部：2011年07月16日至2011年9月8日第四部：2016年暑期档
新还珠格格 ||| 在线播放平台 ||| 芒果TV,PPTV,暴风影音，优酷，搜狐。
新还珠格格 ||| 总策划 ||| 杨文红，苏晓
新还珠格格 ||| 出品人 ||| 欧阳常林
新还珠格格 ||| 总监制 ||| 魏文彬
新还珠格格 ||| entity.description ||| 《新还珠格格》翻拍自琼瑶经典之作《还珠格格》，由李晟、海
```

- Labeling guideline
  1. Show a KB triple to a human annotator;
  2. Let the human annotator to ask a question about this KB triple, whose answer should be the object of the triple.

Summary and Future Work

Summary

- KBQA is crucial to conversational AI
- Existing KBQA methods focus on **one-hop** and **single-turn** questions

SimpleQuestions Nearly Solved:
A New Upperbound and Baseline Approach

Michael Petrochuk
University of Washington Department of Computer Science & Engineering
mikep5@cs.washington.edu

Luke Zettlemoyer
University of Washington Department of Computer Science & Engineering
lsz@cs.washington.edu

- New KBQA datasets with **complex** and/or **multi-turn** questions are appeared

Future Work: KBQA for Complex Questions

Question Type	Question
~~(1) Single-Relation~~	~~when was Steve Jobs born~~
(2) CVT	who played deputy Ferguson in Project Viper
(3) Multi-Hop	which film was written by the director of Wonder
(4) Multi-Constraint	what movie produced by Milan Cheylov and has Samantha Follows acted in it
(5) Multi-Choice	which production is invented by Steve Jobs, Alt code or iPhone
(6) Yes/No	is iPod invented by Steve Jobs
(7) Superlative	what is the longest road in the world
(8) Aggregation	how many children does Bill Gates have
(9) Comparison	which country has more than 100 million people
(10) …	…

Future Work: Conversational KBQA

Who is president of the United States?

Donald Trump

Who are his kids?

Tiffany, Donald Jr., Ivanka, Eric, Barron

When is she born?

Did you mean Tiffany?

No, I mean Ivanka.

October 30, 1981

How about Tiffany?

October 13, 1993