Semantic Web and Linked Big Data Resources for Spoken Language Processing

Dilek Hakkani-Tür and Larry Heck
Microsoft Research, USA

Acknowledgments:
Gokhan Tur, Asli Celikyilmaz

Yun-Nung (Vivian) Chen, Kai Hong, Hongzhao Huang, Yangfeng Ji,
Panupong (Ice) Pasupat, and Lu Wang

TUTORIAL WEB PAGE: http://research.microsoft.com/en-us/projects/kgandld4cu/

Outline

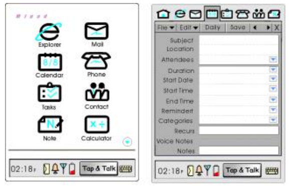
- **Conversational systems**
- Sematic Web and Linked Data Sources
- SLU based on knowledge graphs
- Leveraging Context with knowledge graphs
- Learning from the Semantic Web

Conversational Understanding (CU) Systems

Brief History

TV Voice Search
e.g., Bing on Xbox

Multi-modal systems
e.g., Microsoft MiPad, Pocket PC



2014

Task-specific argument extraction
(e.g., Nuance, SpeechWorks)
User: "I want to fly from Boston to New York next week."

IBM WATSON

Early 2000s

Virtual Personal Assistants:
e.g., Siri, Google now, Cortana

Intent Determination
(Nuance's Emily™, AT&T HMIHY)
User: "Uh…we want to move…we want to change our phone line from this house to another house"

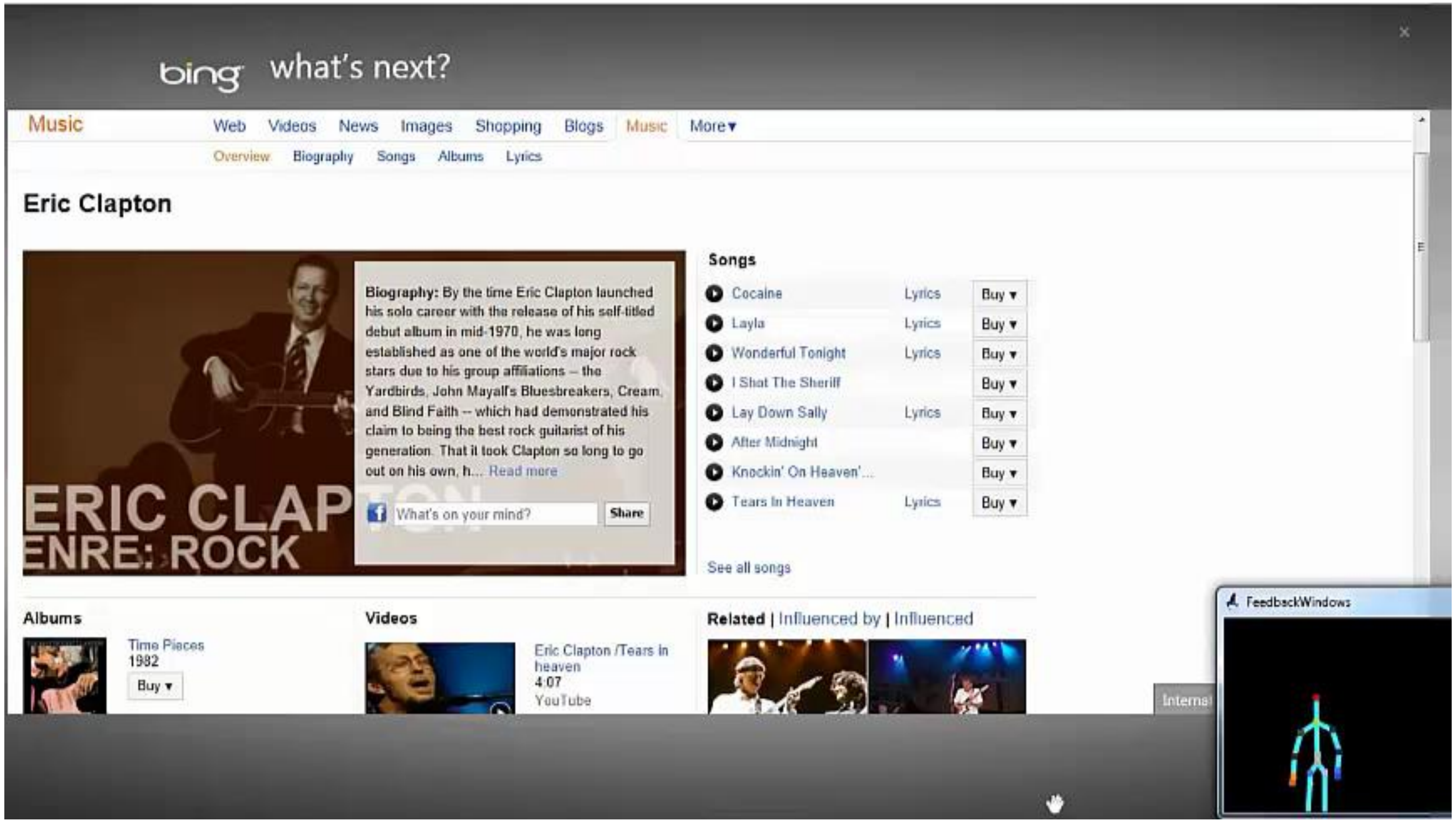
Early 1990s

DARPA
CALO Project

Keyword Spotting
(e.g., AT&T)
System: "Please say collect, calling card, person, third number, or operator"

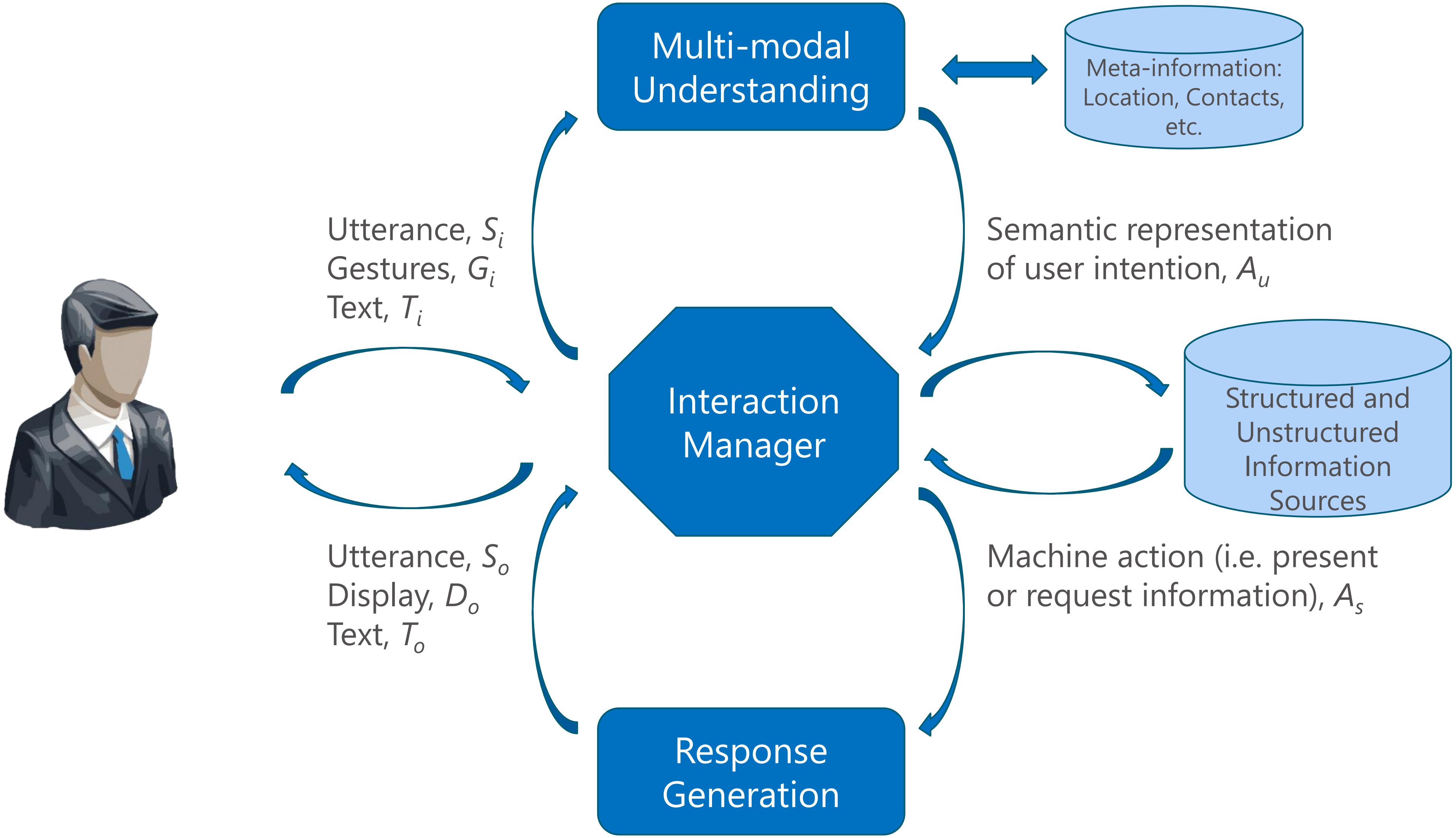
Early Prototype 2012
Conversational Search and Browse

Early Prototype 2012
Conversational Search and Browse



Conversational Understanding (CU) Systems

Conventional Architecture



CU Challenges

- Various ways of saying things
- Non-search-like queries
- Robustness
- Recall/Precision Trade-off
- Meaning Representation

Same request, different wordings:
What is the name of the actor that played Jake Sully in Avatar
Who played Jake in Avatar
What was the actor who played Jake in Avatar
Find the name of the actor who played Jake Sully in Avatar
Who was the actor that played Jake Sully in Avatar
Find the guy who played Jake Sully in Avatar
Who is the guy who played Jake Sully in Avatar
Actor who played Jake Sully in Avatar
What was the name of the actor who plays Jake Sully in Avatar
Actor from Avatar who played Jack Sully
Can you tell me the name of the actor who played Jake in Avatar
I need to know the name of the actor for Jake Sully in Avatar
Show me Sully from Avatar
I need to know the real name of Jake Sully on Avatar
Who is Jake Sully in Avatar
Give me the name of Jack Sully on Avatar
….

CU Challenges

- Various ways of saying things
- Non-search-like queries
- Robustness
- Recall/Precision Trade-off
- Meaning Representation

More common in:

Keyword queries → Search

Starbucks
Pandora
…

NL-like queries → Conversational Interactions

Who is the director of Avatar movie
What are the common symptoms of …
…

CU Challenges

- Various ways of saying things
- Non-search-like queries
- Robustness
- Recall/Precision Trade-off
- Meaning Representation

- **ASR errors**

  who plays jake sunny in avatar

- **Disfluencies**

  how about uhm some some th- uhm no italian ones

- **Ungrammatical utterances**

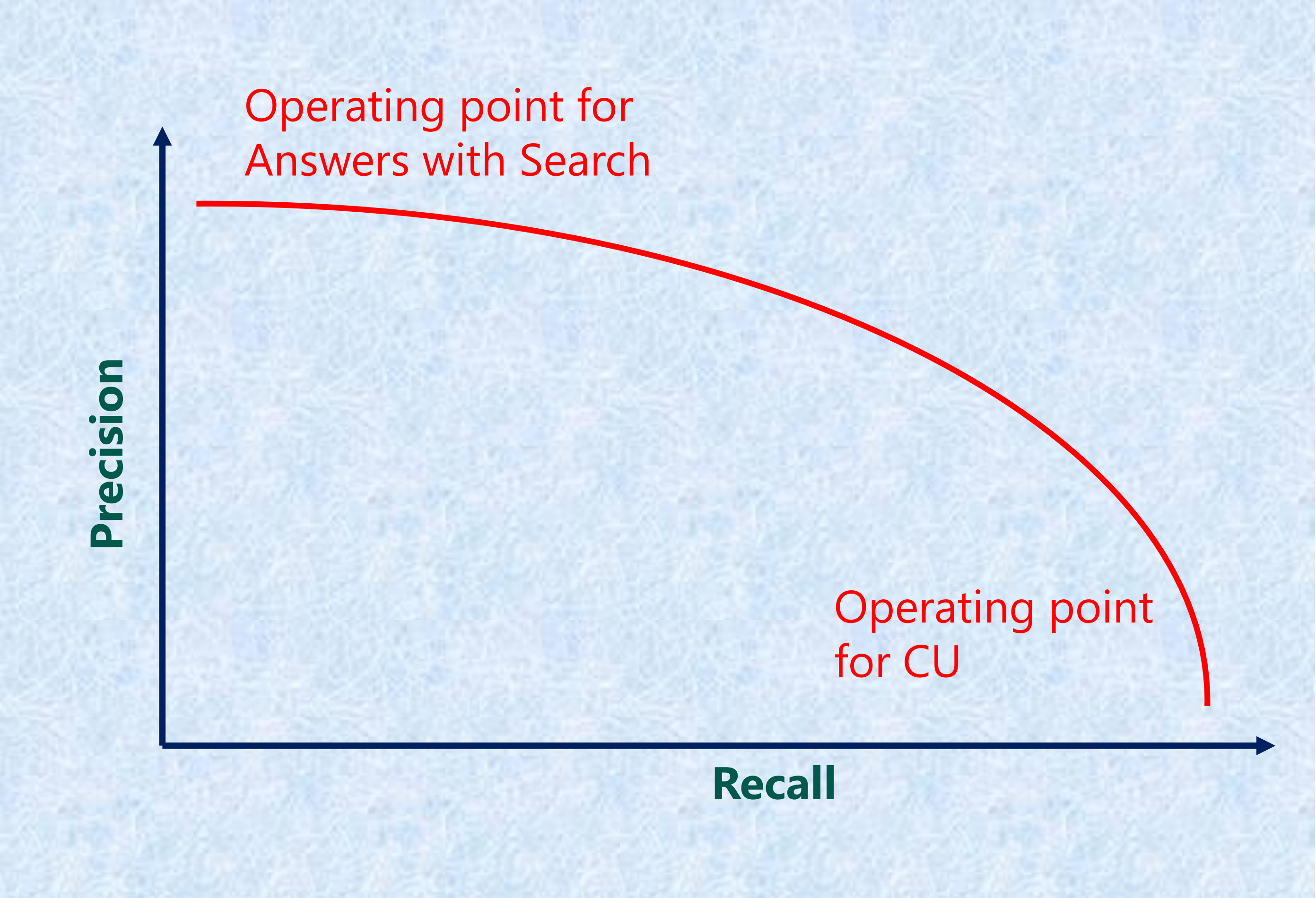
  who was who actor that played jake sully in avatar

- **Lack of capitalization/punctuation**

  i don't like this why don't you show me ones like uh avatar

CU Challenges

- Various ways of saying things
- Non-search-like queries
- Robustness
- Recall/Precision Trade-off
- Meaning Representation



CU Challenges

- Various ways of saying things
- Non-search-like queries
- Robustness
- Recall/Precision Trade-off
- Meaning Representation

- **Task Independent Representations**
  - Examples: PropBank, FrameNet, AMR
  - Open domain coverage
  - Require mapping to knowledge sources
- **Task-dependent Representations**
  - Examples: ATIS schema
  - Schema is manually designed by domain experts.
  - High coverage for specific domains

Spoken Language Understanding (SLU)

- **Semantic Representation:** Flat or hierarchical frame of domain, intent, and slots.

DOMAIN = movies
"when was james cameron's avatar released"

INTENT: Find_release_date
MOVIE NAME: avatar
DIRECTOR NAME: james cameron

Intents
Find movie
Find showtime
Find theater
Buy tickets
…

Slots
Movie genre
Movie award
Theater location
Number of tickets
…

DOMAIN = company
"show me media companies in california"

INTENT: Find_company
LOCATION: california
INDUSTRY: media

Intents
Find company
Find revenue
Find founder
Find contact
…

Slots
Company name
Company address
Company revenue
Company industry
…

Domain/Intent Classification

- Mainly viewed as utterance classification.
- Given a collection of labeled utterances:

$$D = \{(u_1, c_1), \ldots, (u_n, c_n)\}$$

where $c_i \in C$, the goal is to estimate

$$c_k' = \operatorname{argmax}_{c \in C} P(c|u_k)$$

Example, "Show me the nearest movie theater"

Domain: movies

Intent: find-theater

Domain/Intent Classification

- Heavily influenced by call routing applications for customer care centers
  - AT&T HMIHY: Using salient phrases (Gorin et al., 1997), boosting (Gupta et al., 2006), Support Vector Machines (Haffner et al., 2003), extended boosting by using prior knowledge (Schapire et al. 2005)
  - Bell Labs: Vector space model (Chu-Carroll and Carpenter, 1997), extended by MCE/Boosting by Kuo, Lee, Zitouni, et al., 2000 – 2003
  - BBN Call Director (Natarajan et al., 2002)
  - France Telecom 3000 (Damnati et al., 2007)
  - (Cox et al., 2003), extended by linear discriminant analysis (LDA)
  - DCNs, (Tur et al., 2012a)
- Tighter integration with ASR:
  - Using word confusion networks (Hakkani-Tür et al., 2006), language model adapted by the intent model (Riccardi et al., 1996 and Chelba et al., 2003), language model discriminatively trained using MCE (Yaman et al., 2008).

Slot Filling

- Word sequence classification
- Given a collection tagged word sequences,

$$S=\{(w_1,t_1),\ldots,(w_n,t_n)\},$$

where $t_i = t_{i,1},\ldots,t_{i,|ui|}$, and $t_{i,m} \in M$, the goal is to estimate

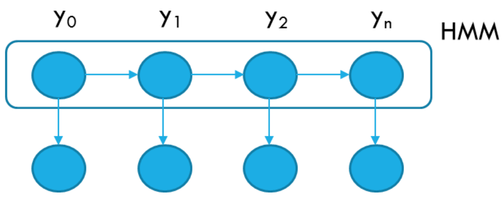
$$t_k' = \operatorname{argmax}_t P(t|w_k)$$

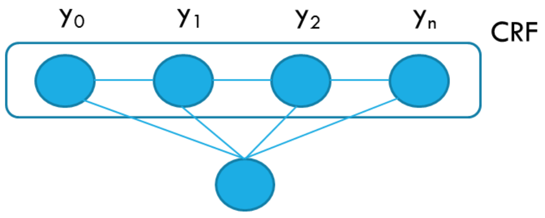
Example:

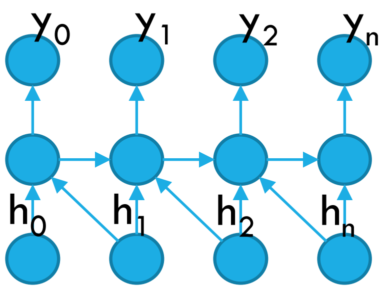
flights	from	Boston	to	New	York	today
O	O	B-city	O	B-city	I-city	O
O	O	B-dept	O	B-arrival	I-arrival	B-date

Slot Filling (cont.)

- Knowledge-Based
  - CMU Phoenix (Ward et al., 90s), MIT – TINA (Seneff et al., 90s), Apple Siri (Cheyer et al.), SRI Gemini (Appelt et al. 90s), Amazon Evi (TrueKnowledge), Nuance Vlingo
- Statistical
  - Generative Models: AT&T – Chronus (Levin&Pieraccini, 90s), BBN – Hidden Understanding Models (Miller et al., 90s), Cambridge (Young and He, 2004), U. Avignon (using FSM framework)
  - Discriminative Models: McGill – Chanel (De Mori et al., 90s), LUNA (Riccardi and Raymond, 2006), Microsoft (Wang et al., Tur et al., Mesnil et al., Yao et al., etc.)







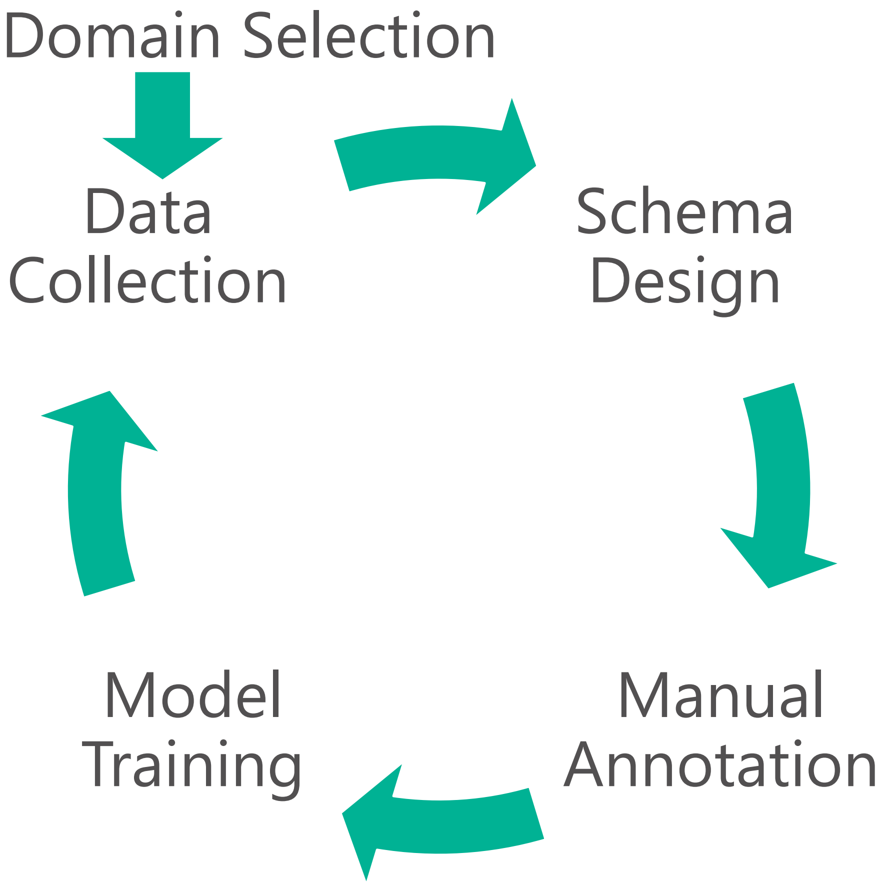
Slot Filling (cont.)

- Hybrid
  - Fewer high precision rules, rest handled by statistical models.
  - Nuance, Maluuba (probably), U. Aachen (using MT framework)
- Tighter integration with ASR
  - Joint decoding of ASR and semantic tagging (Deoras et al., 2013), semantic parsing using word confusion networks and CRFs (Tur et al, 2013).

Joint Modeling of Intent and Slots

- Triangular CRF (Jeong and Lee, 2008)
- Convolutional neural network based triangular CRF (Xu and Sarikaya, 2014).
- Intent as one of the slots (Kennington et al, 2013)

Conventional SLU Life Cycle



As a result…

Motivation

- As a result…
  - Narrow breadth of domains
  - Limited sharing of data/schemas between domains
  - Limited ability to incorporate disparate knowledge sources
  - Inflexible to changes in task definition
  - Requirement of mapping to the back-end (interpretation)

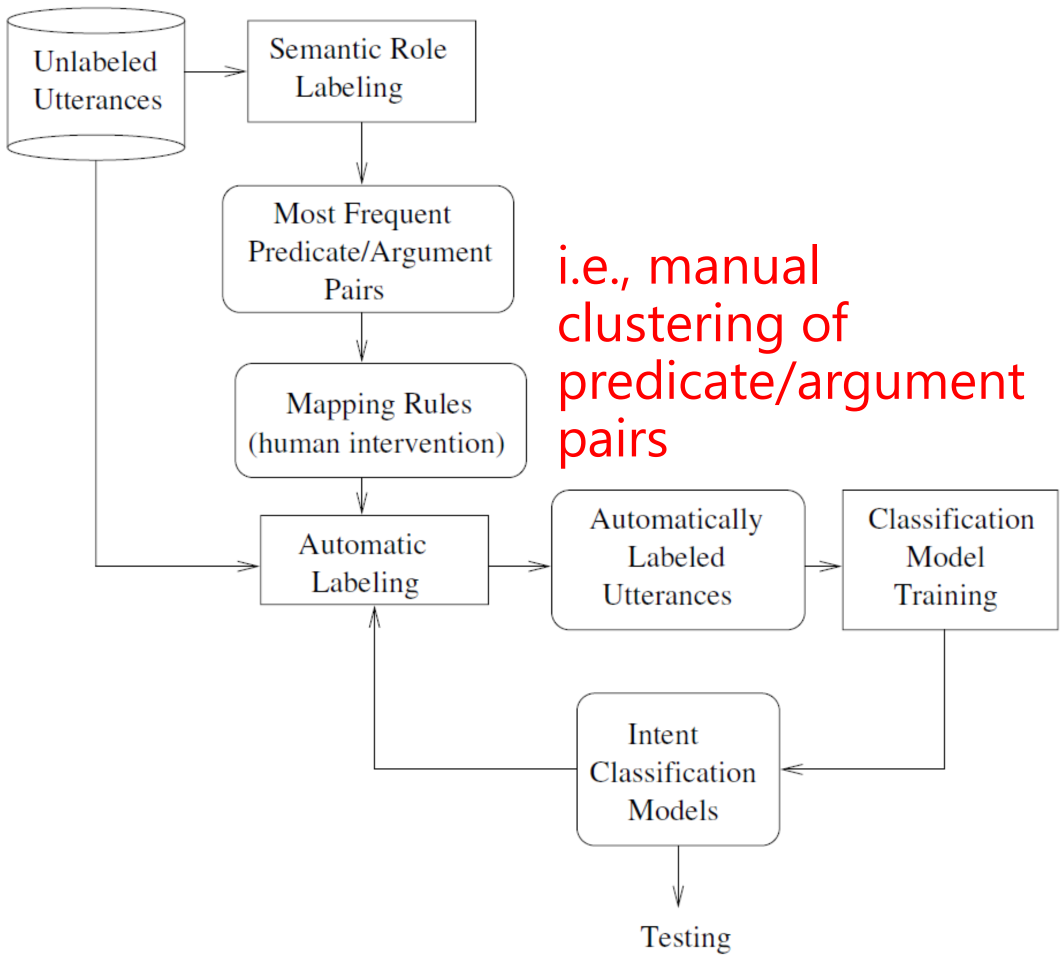
- How to reduce time to create and deploy SLU for a given domain?
- How about many domains?
- Can we exploit web search resources and linked big data?

Earlier work: Discovering Domain Concepts
(Chotimongkol & Rudnicky; 2002)

- Hierarchical clustering algorithms to merge similar words or clusters, according to their distribution in goal-oriented human-human conversations.
  - Mutual Information
  - Kullback-Leibler Divergence
- Quantitative Evaluation: Recall and Precision against reference concept labels.
- CMU Travel Domain corpus (hotel, flight, car), 50-60% F-measure.

Earlier work: Semantic Role Labeling (PropBank)

(Tur, Hakkani-Tur, Chotimongkol, 2005)



i.e., manual clustering of predicate/argument pairs

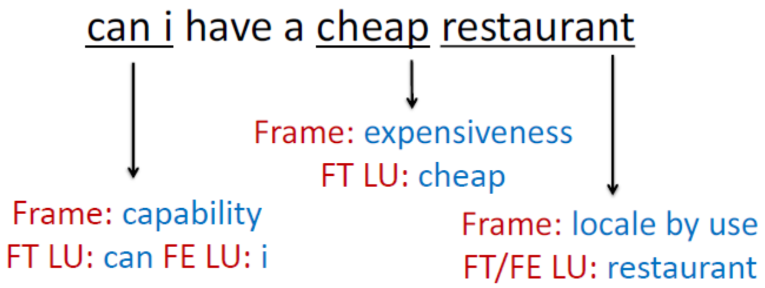
Pred./Arg. pair, p	**Arg. Type**	**Call-type, c**	$P(p\|c)$	$P(c\|p)$
place/order	*Arg1*	Make(Order)	0.77	0.96
make/order	*Arg1*	Make(Order)	0.03	0.93
order/something	*Arg1*	Make(Order)	0.02	0.86
check/order	*Arg1*	Check(Order_Status)	0.14	0.95
cancel/order	*Arg1*	Cancel(Order)	0.07	0.95
check/status	*Arg1*	Check(Order_Status)	0.50	1.00
talk/someone	*Arg2*	Talk(Human)	0.05	0.89
talk/somebody	*Arg2*	Talk(Human)	0.5	0.91

- Semi-supervised method achieved 86.5% of the performance of a classification model trained with thousands of labeled utterances.
- Experimented with data from 2 domains.

Earlier work: Semantic Parsing (FrameNet)

[Chen, Wang, Rudnicky, 2013]

- Unsupervised induction and filling of semantic slots.
- SEMAFOR, FrameNet Semantic Parser to generate initial semantic parses from ASR output.
- Slot ranking model based on frequency and coherence, to identify generic semantic versus domain-specific concepts.



- Coherence: based on spectral clustering and distribution similarity of slot values over clusters.
- CU restaurant corpus, unsupervised slot filling F-meas.: ~30%

Active Learning

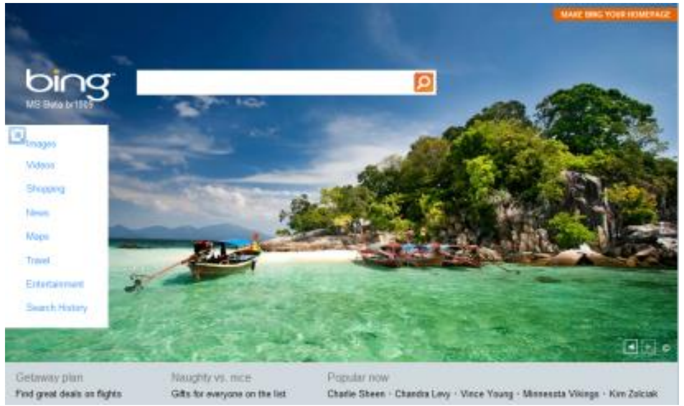
- Aim to reduce the amount of data to be labeled to obtain the same performance with less data, by focusing on informative examples.
- [Tur, Hakkani-Tur, Schapire; Speech Communication, 2005], certainty based active learning method that selects examples with low classifier scores for manual annotation.
- 25% of the training examples result in the same error rate with all the data.
- Extended active learning framework that combines output from multiple systems via ROVER [Sarikaya; Speech Communication, 2008]

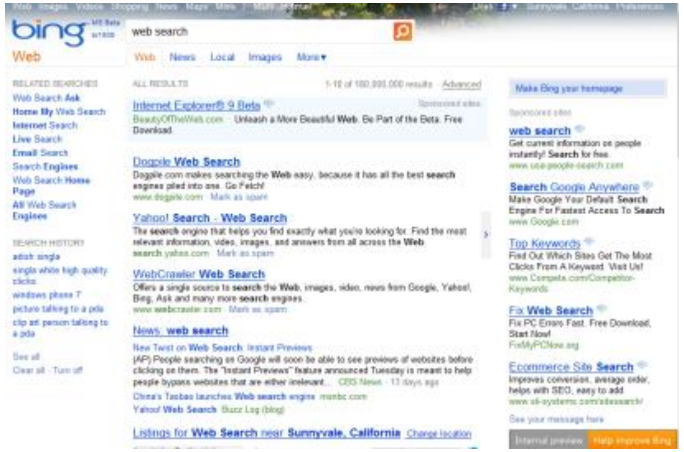
Active and Unsupervised Learning

- Manually label informative examples
- During training, use unlabeled examples with automatically extracted labels [Tur, Hakkani-Tur, Schapire; Speech Communication, 2005]:
  - Merge the two data sets for training
  - Model adaptation

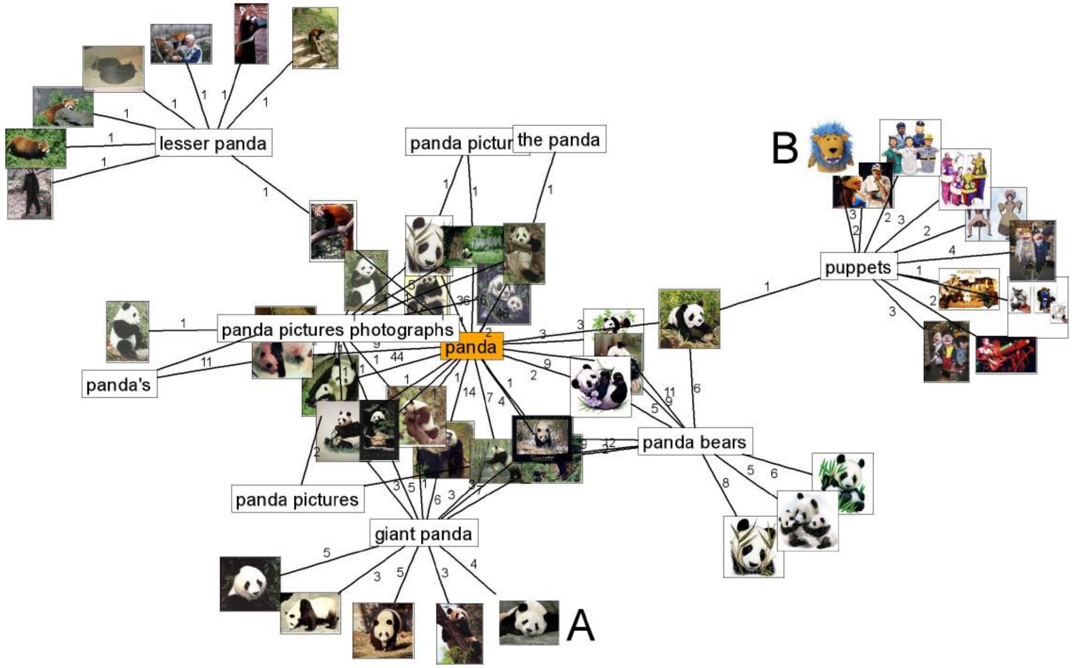
Web Search and Query Click Logs

- Include:
  - Queries
  - URLs returned by the search engines and clicked by the users
- Possible to extract information from
  - Multiple users' behavior
    - For mining high-quality query and click pairs (White and Singla, 2010)
  - Search sessions
    - Users' reformulation of their queries.
    - Modeling interactions, sequencing of intents.
    - Dwell times, etc.





Query Click Logs represented as a Graph



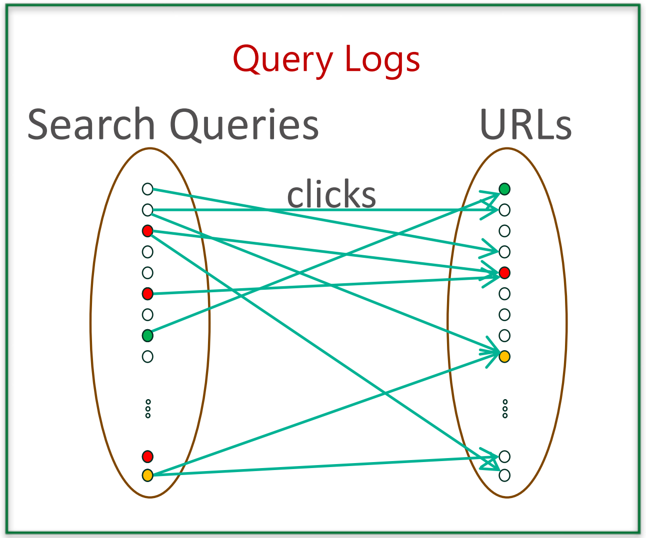
- Click information modeled as graph (Craswell & Szummer, 2007)
- Two types of nodes: queries and documents (represented by URLs)
- An edge connects a query and a document if we have observed a click for that query-document pair by any user.
- Edges can also be weighted by frequencies.

Query Click Logs: Bipartite Graph

Calories in double chocolate chip mocha

weather in los altos

Malaria symptoms



www.yummly.com/recipes/double-chocolate-chip-mocha

www.weather.com/weather/today/Los+Altos+CA+94022

en.wikipedia.org/wiki/Malaria

The graph can be used to find associations between nodes that are not directly linked, by following paths in the graph:

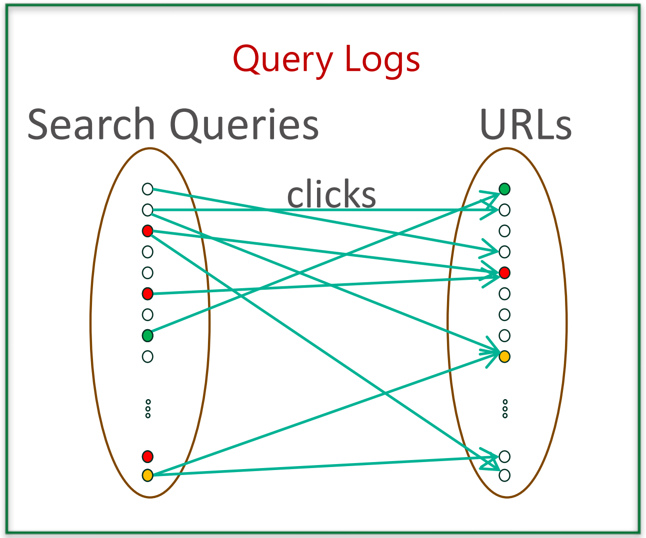
- Query to document search
- Query to query suggestion
- Document to query annotation
- Document to document relevance feedback

Query Click Logs: Bipartite Graph

Calories in double chocolate chip mocha

weather in los altos

Malaria symptoms



www.yummly.com/recipes/double-chocolate-chip-mocha

www.weather.com/weather/today/Los+Altos+CA+94022

en.wikipedia.org/wiki/Malaria

Edges can be weighted with transition probabilities, for random walk algorithms.

$$P_{t+1|t}(k \mid j) = \begin{cases} (1-s)C_{jk}/\sum_i C_{ji} & \forall k \neq j \\ s & \text{when } k = j. \end{cases}$$

i ranges over all nodes

s self transition probability

C_{jk} click counts associating node j to k

CU versus Search queries

Frequencies of various query forms

Search Queries

Template	Frequency
ent	44.9%
$type \sqcap rel(ent)$	12.8%
$ent_0 \sqcap rel(ent_1)$	7.7%
$ent \sqcap type$	5.8%
$type$	5.8%
$attr(ent)$	3.8%
$ent_1 \sqcap rel(ent_0)$	3.2%
$rel(ent)$	1.9%
$ent_0 \sqcap rel(ent_1, rel(ent_2))$	1.3%
$type_1 \sqcap rel(type_0)$	1.3%

Ten most frequently occurring templates among entity-based queries (Pound et al., CIKM'12)

CU Queries

Template	Rel. Frequency
No. with SPARQL annotations	3,338
% with no relation (i.e. entity only)	10.1%
% with 1 relation	70.4%
% with 2 relations	10.2%
% with 3 or more relations	1%
% not covered by Freebase	8.3%

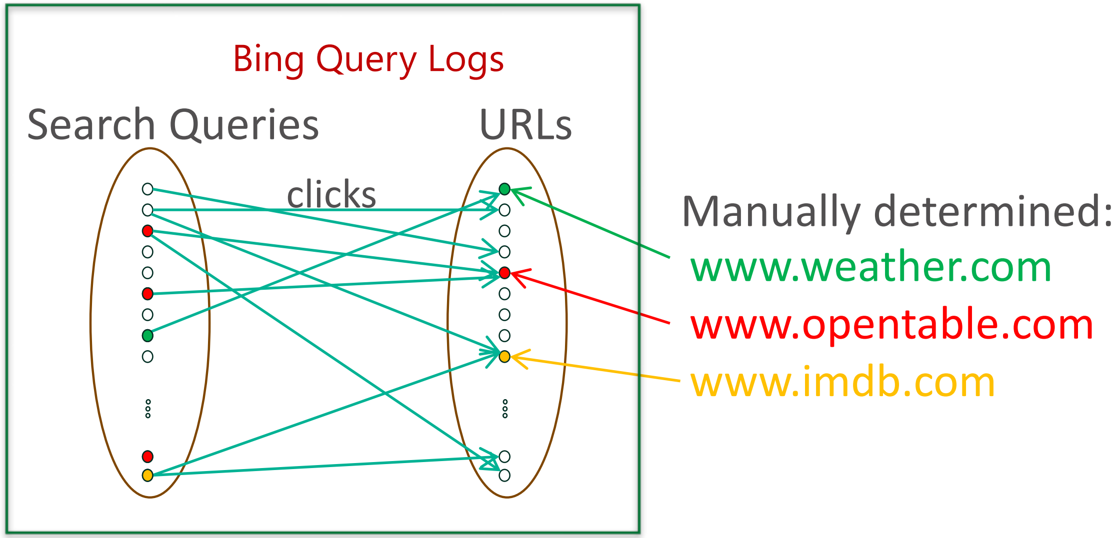
We estimate 3-4% of search queries to be NL-like.

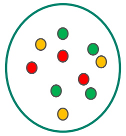
Earlier Work Based on Search Query Click Logs

Unsupervised:

- 18.9% to 7.9%

[HakkaniTurEtAl, 2011a]





Manually labeled SLU queries

e.g.: x = "I wanna reserve a table at Zucca."

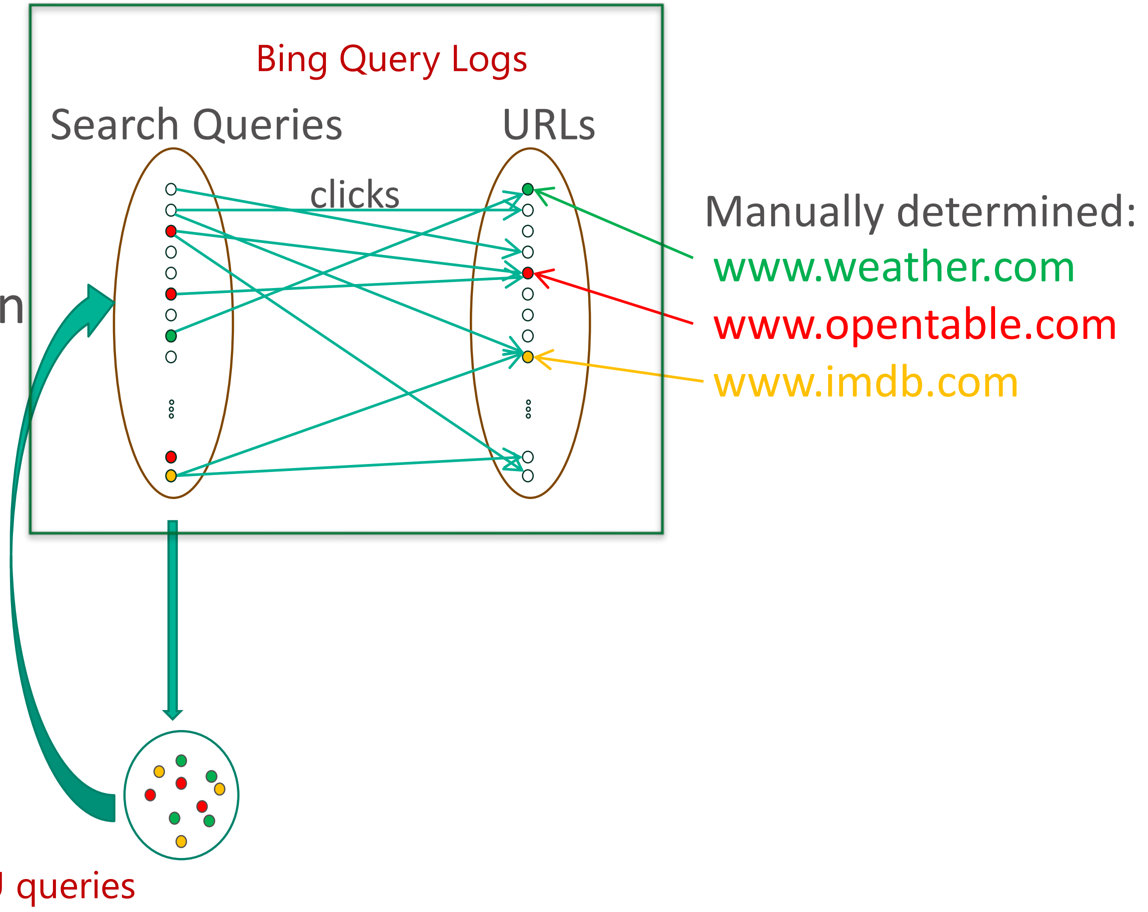
y = "restaurant"

Earlier Work Based on Search Query Click Logs

Semi-Supervised:

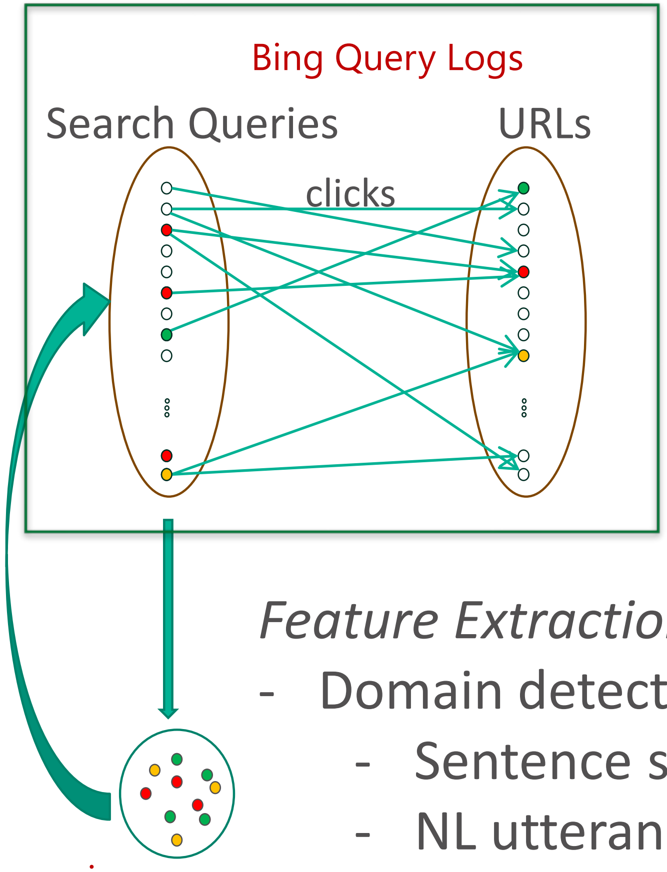
- Self Training
- Noisy Label Propagation
- 6.2% to 5%

[HakkaniTurEtAl, 2011b]



e.g.: *x* = “I wanna reserve a table at Zucca.”
y = “restaurant”

Earlier Work Based on Search Query Click Logs



e.g.: x = "I wanna reserve a table at Zucca."
y = "restaurant"

NL Search Query (*DISP are indented*)	Keyword Query
what are the signs of throat cancer	throat cancer symptoms
what are the biggest us companies	fortune 500 companies

Feature Extraction:

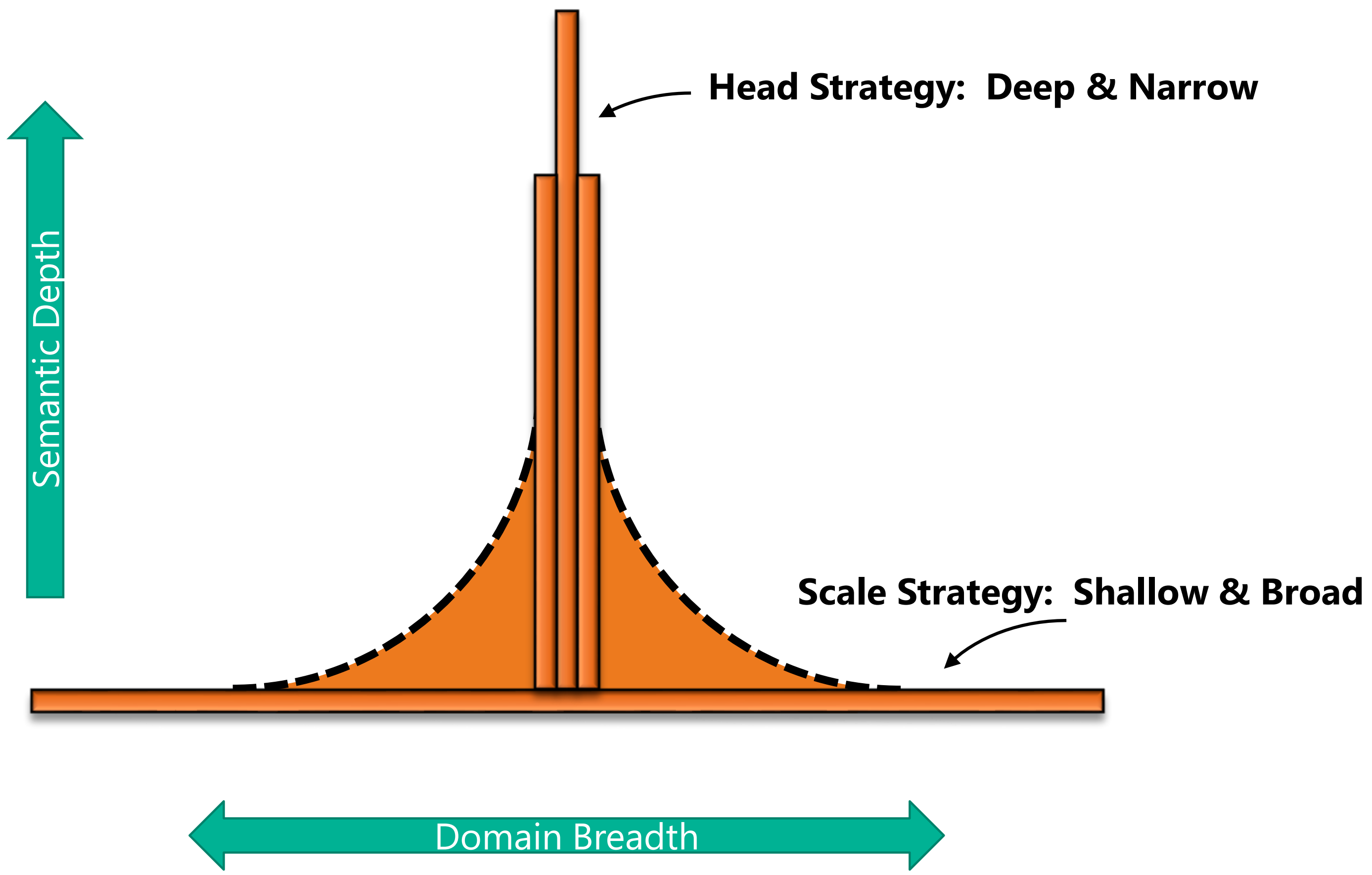
- Domain detection error rate reduction (from 10.6% to 5.4%):
  - Sentence simplification with syntax[TurEtAl, 2011],
  - NL utterance to query translation [HakkaniTurEtAl, 2011c]
  - Kernel-DCN [TurEtAl, 2012; DengEtAl, 2012]
  - Zero-shot DNN [DauphinEtAl, 2014]

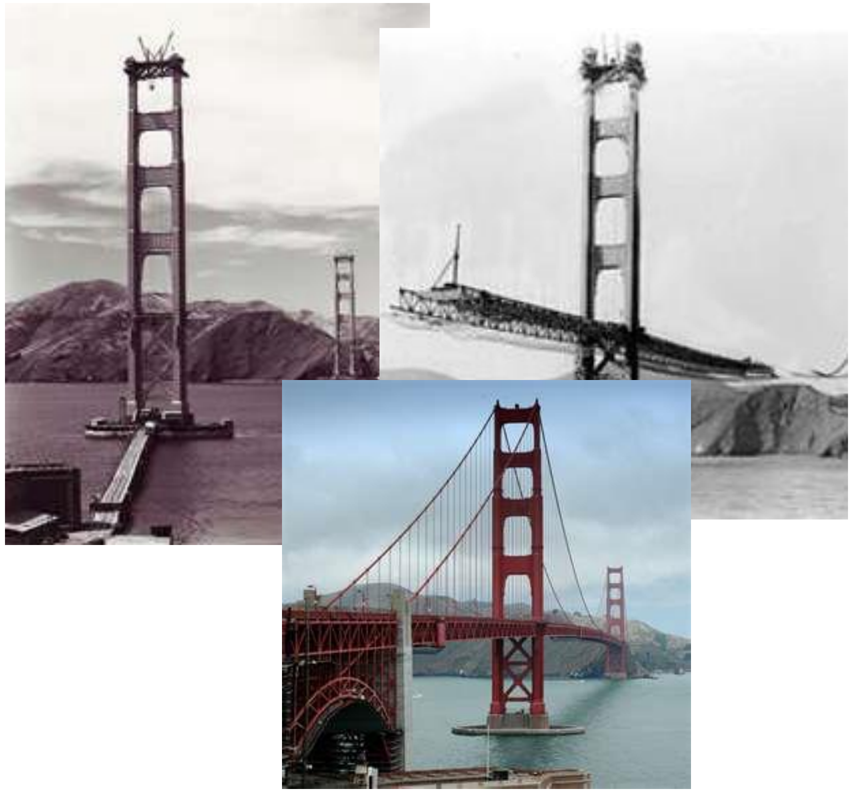
Outline

- Conversational systems
- **Sematic Web and Linked Data Sources**
- SLU based on knowledge graphs
- Leveraging Context with knowledge graphs
- Learning from the Semantic Web

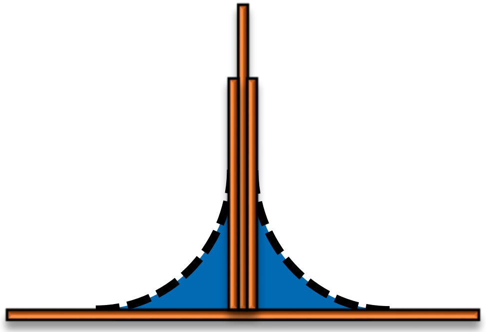
Conversational Systems Challenge

Scaling *Depth and Breadth*





The Opportunity



Knowledge is the Foundation of Conversations

A vast majority of user interactions are with people, locations, things (**entities**).

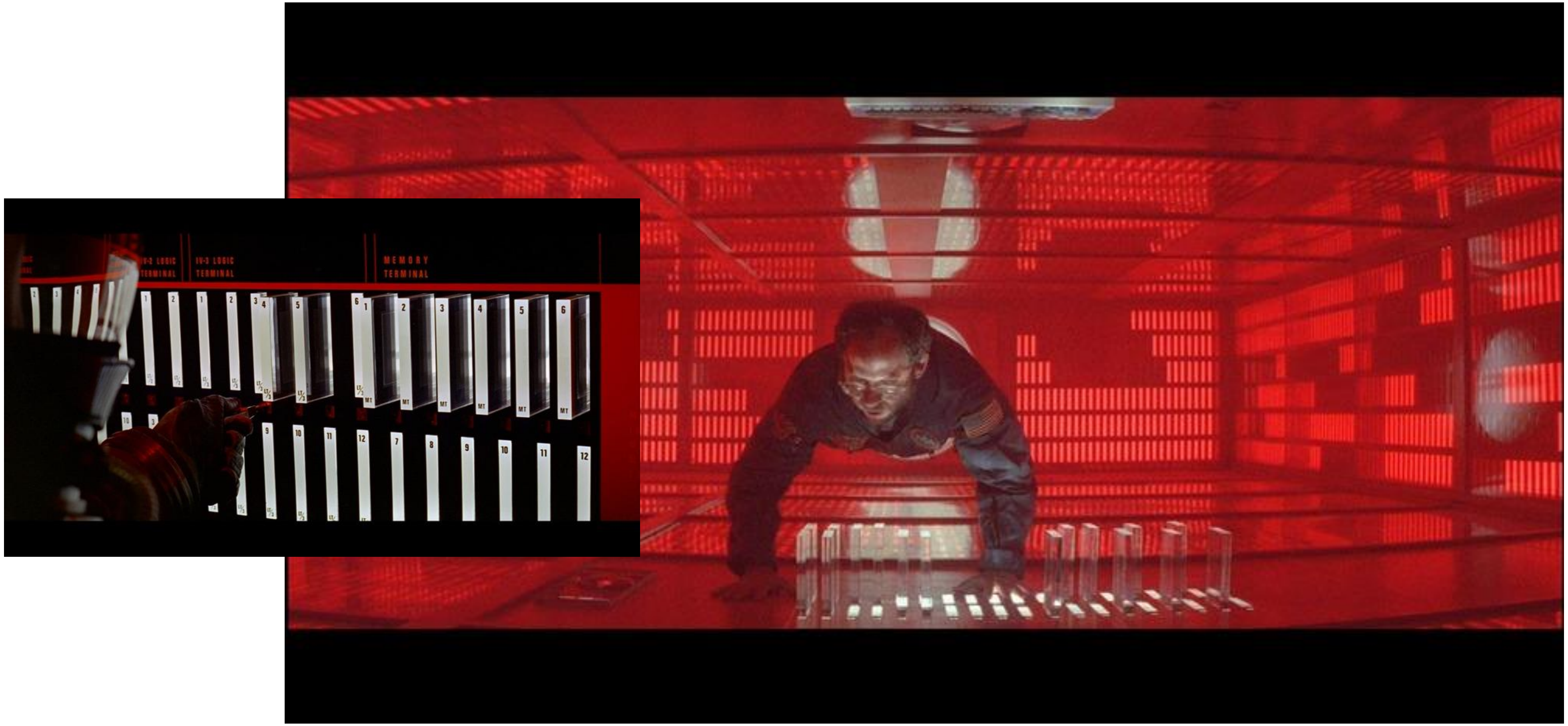
Knowledge refers to these **entities**/concepts and to how they are interrelated.

The dual-role of knowledge

- People seek to **browse** and **find information** about **entities** and to **transact** on them.
- Knowledge serves as a **grounding for conversations**.

Knowledge "Crystals"

Vision: *Push-button NUI from Knowledge Graph*



Knowledge Graphs

What are knowledge graphs?

Graphs of strongly typed and uniquely identified **entities** (nodes) and facts/literals connected by **relations** (edges)

Examples

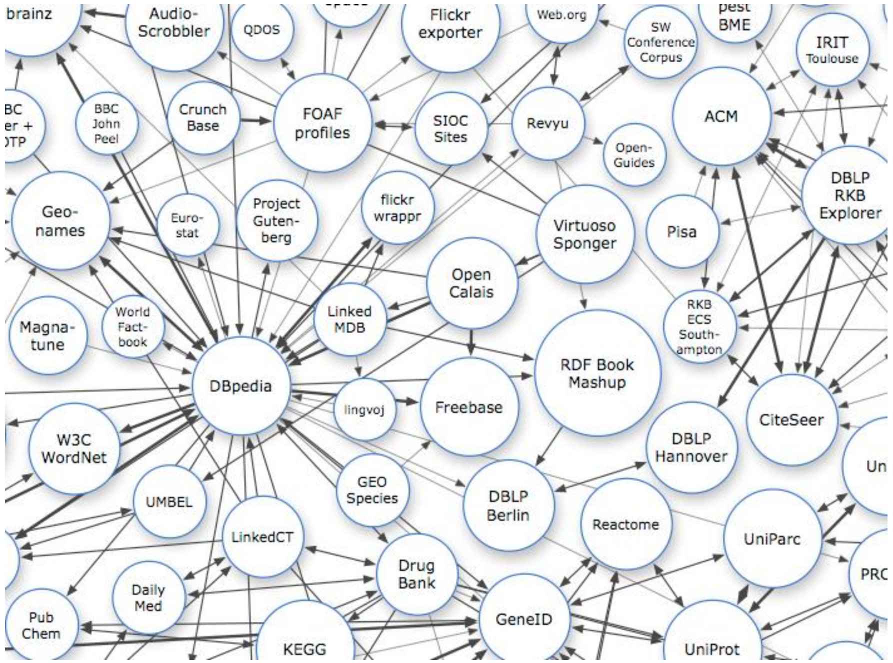
Satori, Google KG, Facebook Open Graph, Freebase

How large?

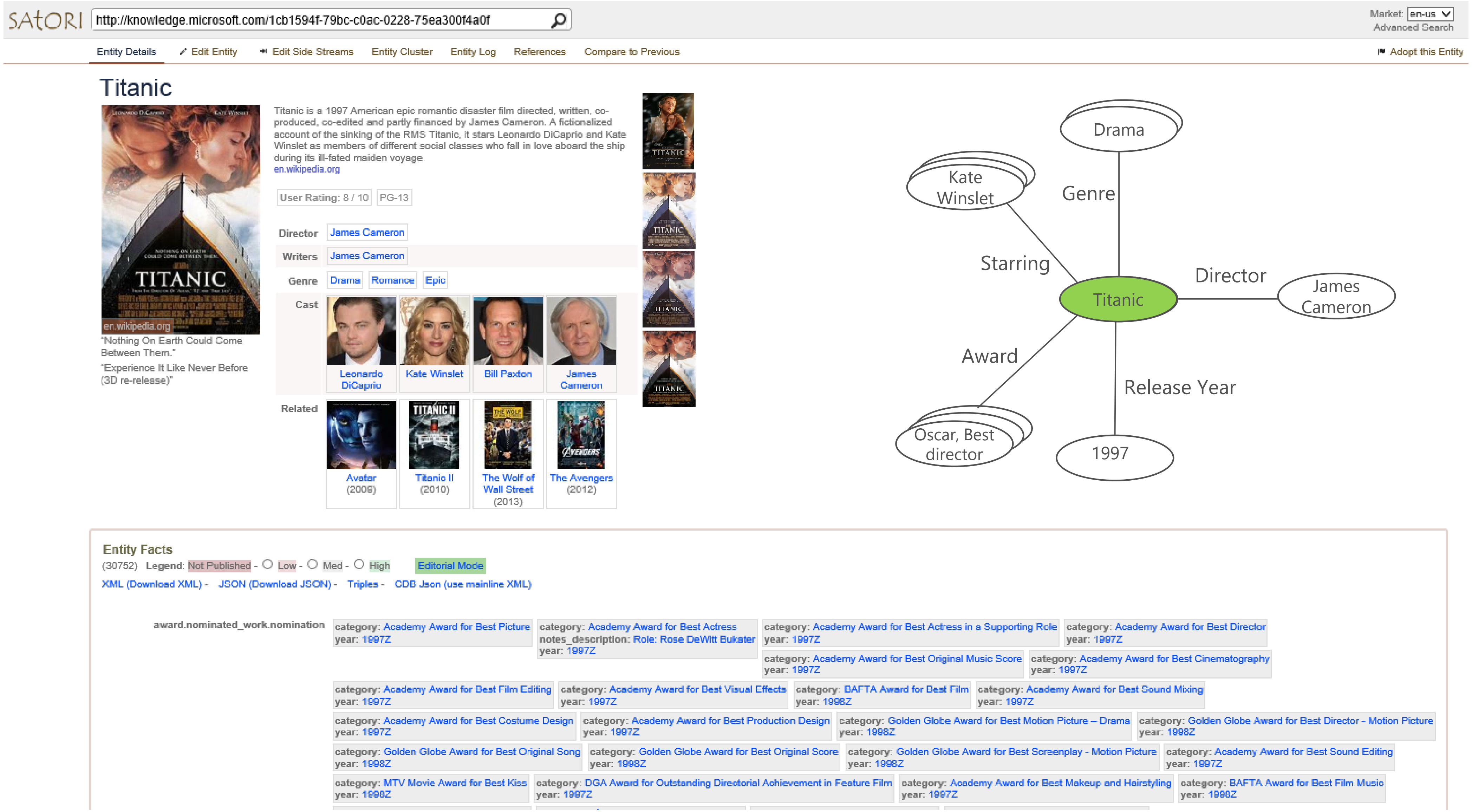
> 500M entities, >1.5B relations, >5B facts

How broad is the knowledge?

Wikipedia-breadth: "American Football" ← → "Zoos"



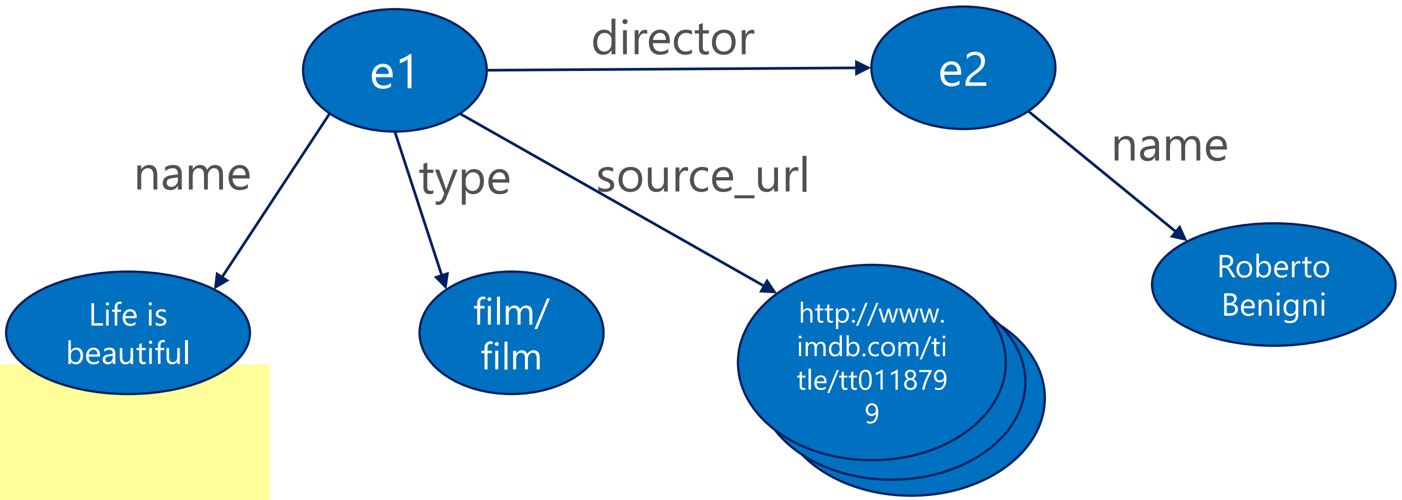
Knowledge Graphs



Knowledge Graphs

- RDF: Resource description framework, W3C standard for encoding knowledge.
- Graph structures to represent and store knowledge
- Graph is represented as "triples":

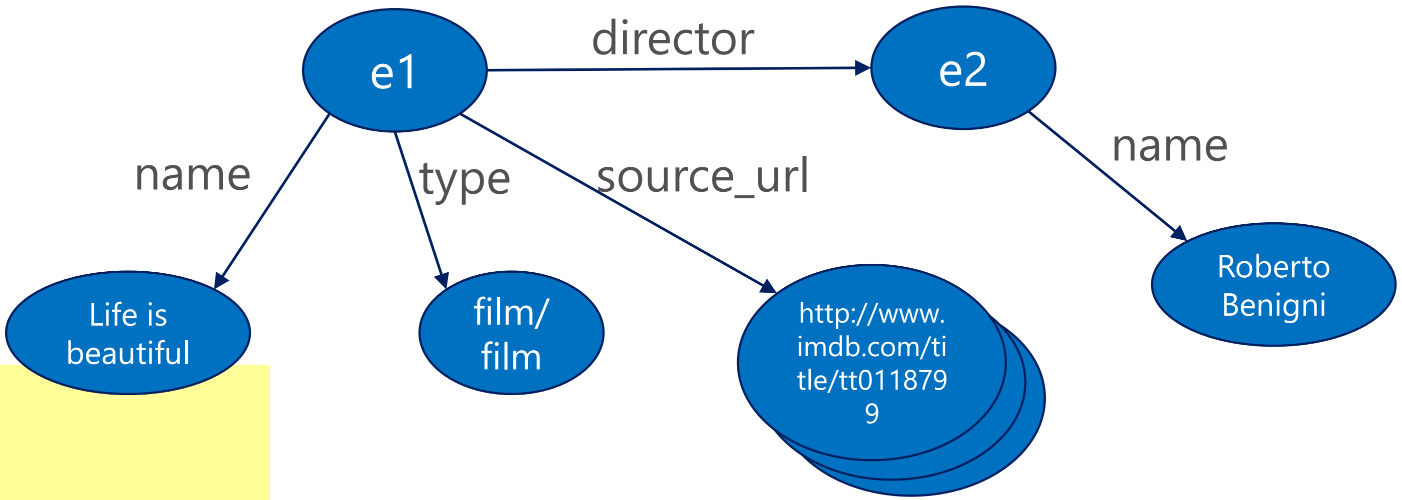
Subject, Predicate, Object



e1 name "Life is beautiful".
e1 type film/film.
e1 source_url http://www.imdb.com/title/tt0118799.
e1 director e2.
e2 name "Roberto Benigni".

Knowledge Graphs

- Two types of triples:
  - Connecting entities to other entities.
  - Connecting entities to their attributes.

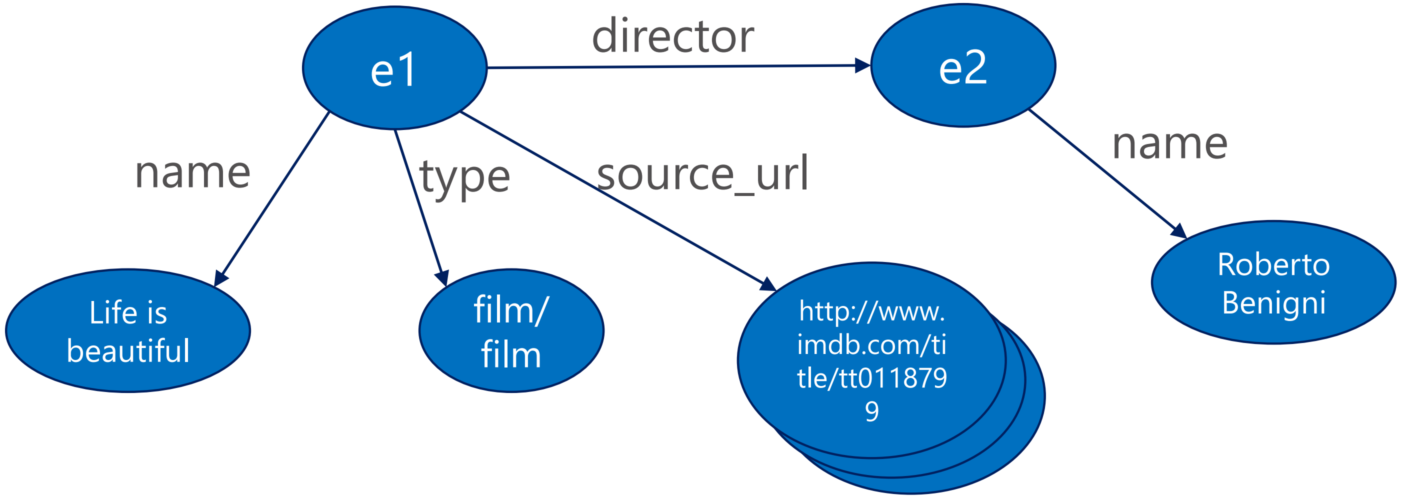


e1 name “Life is beautiful”.
e1 type film/film.
e1 source_url http://www.imdb.com/title/tt0118799.
e1 director e2.
e2 name “Roberto Benigni”.

Knowledge Graphs (cont.)

- Can represent knowledge in multiple languages.

e1 name "Life is beautiful":EN.
e1 name "La vita è bella":IT.

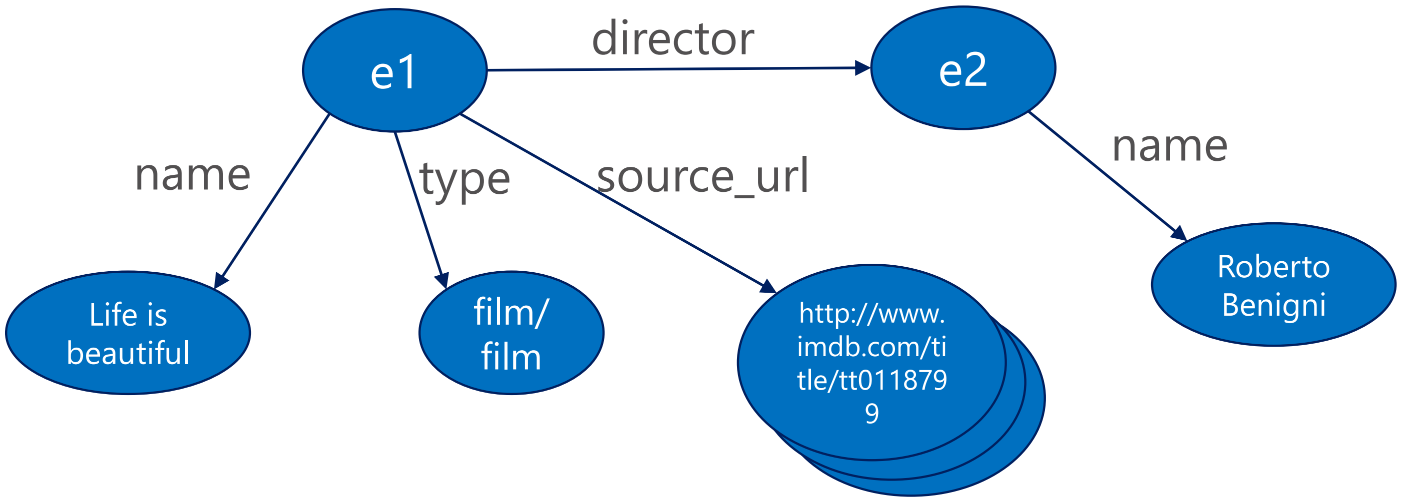


Knowledge Graphs (cont.)

- Can relate entities with multiple attribute values/entities:

e3 name “Brad Pitt”:EN.
e3 type film.actor.
e3 type film.director.

e2 source_url
http://www.imdb.com/title/tt0118799.
e2 source_url
http://movies.msn.com/movies/movie/life-is-beautiful.1/.



Knowledge Graphs (cont.)

- Populated with knowledge and growing:

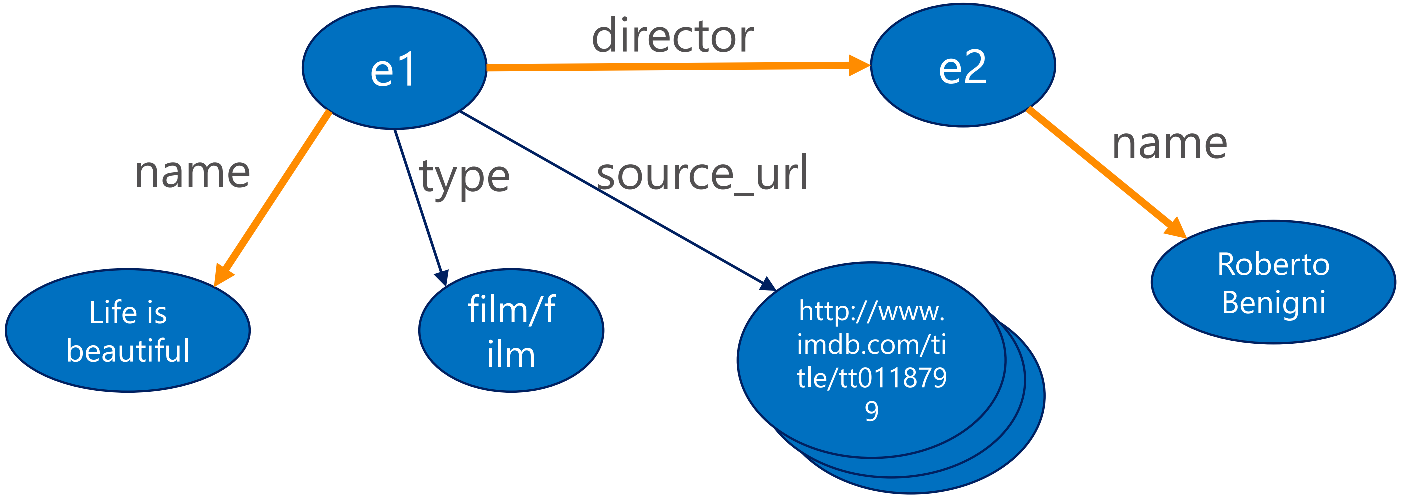
11.5.2013	Freebase
# of domains	92
# of types	~2 K
# of relations	~5 K
# of entities	~37 M
# of triples	~683 M

- Source URLs and other related URLs allow for joining with
  - Wikipedia documents
  - Search query click logs
  - Dump of the whole web text (in petabytes)

SPARQL

- Query language for querying knowledge graphs.

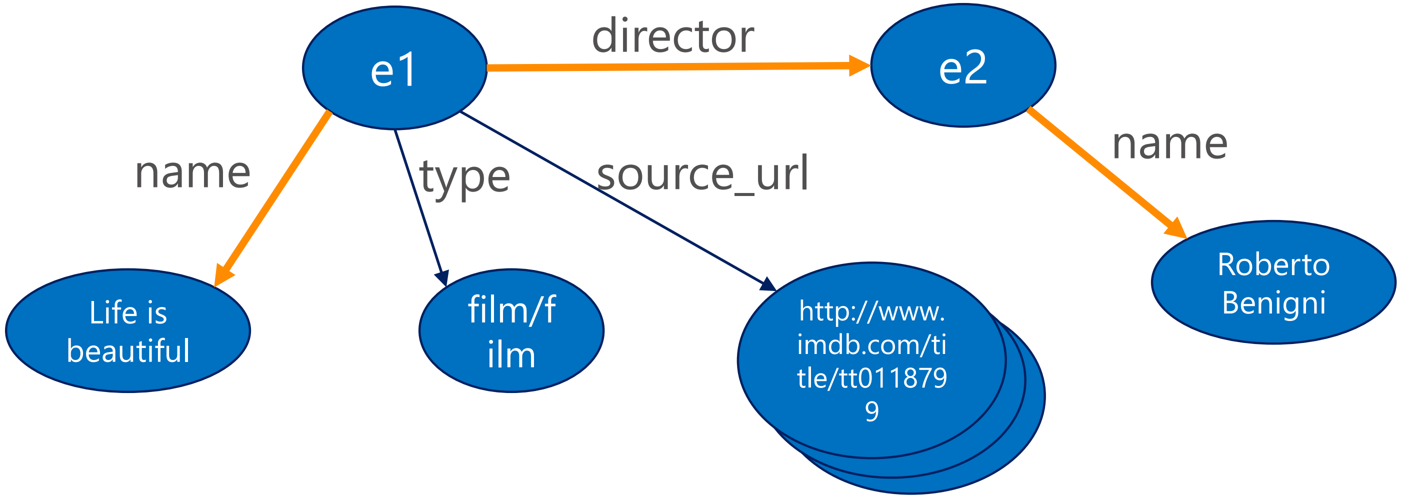
```
SELECT ?dname WHERE {
    ?movie name “Life is beautiful”.
    ?movie director ?director.
    ?director name ?dname.
}
```



SPARQL

- Each query is a sub-tree of the graph.

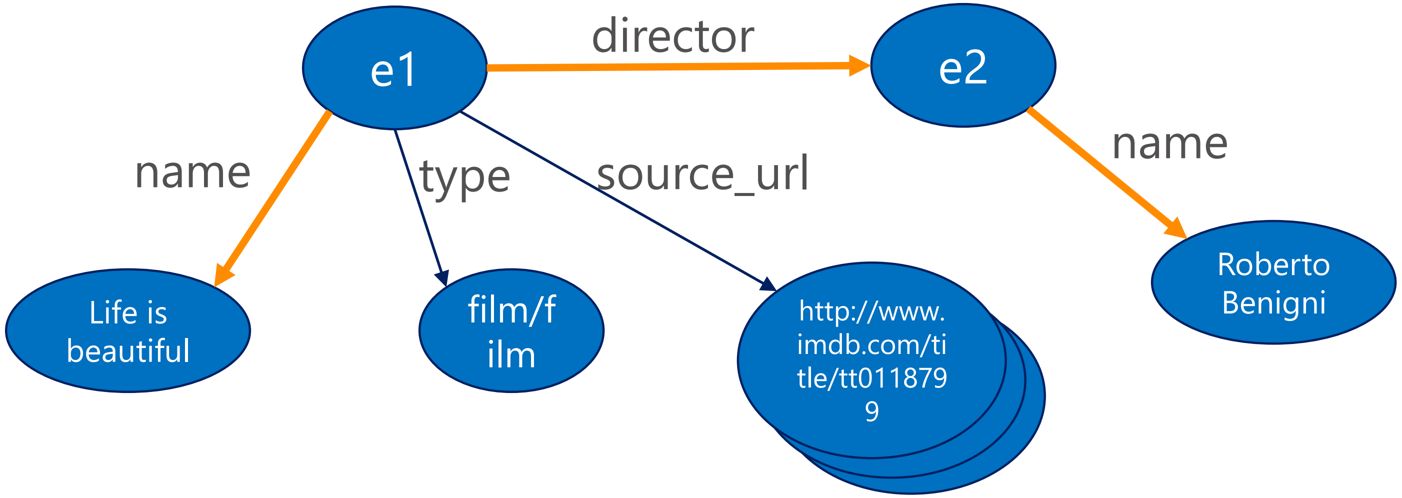
```
SELECT ?dname WHERE {
    ?movie name “Life is beautiful”.
    ?movie director ?director.
    ?director name ?dname.
}
```



SPARQL

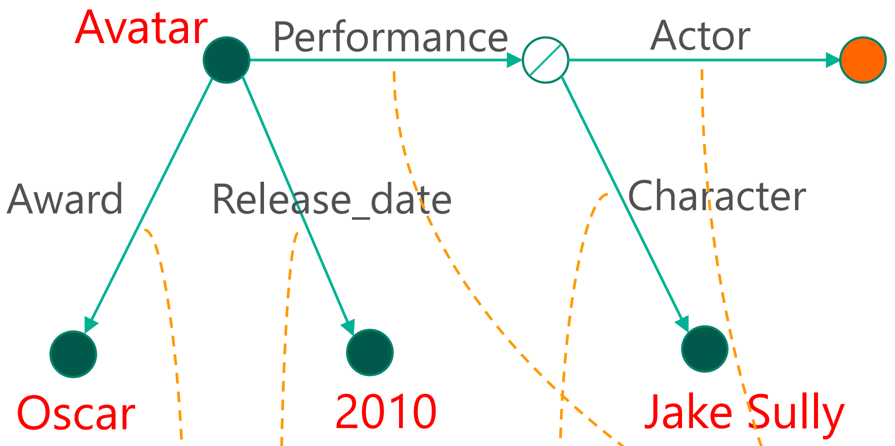
- Querying graph patterns along with their conjunctions and disjunctions.
- Also supports aggregation, sorting, negation, etc.

```
SELECT ?dname WHERE {
    ?movie name “Life is beautiful”.
    ?movie director ?director.
    ?director name ?dname.
}
```



Example 1

U1: Who played Jake Sully in the Oscar winning 2010 movie Avatar?



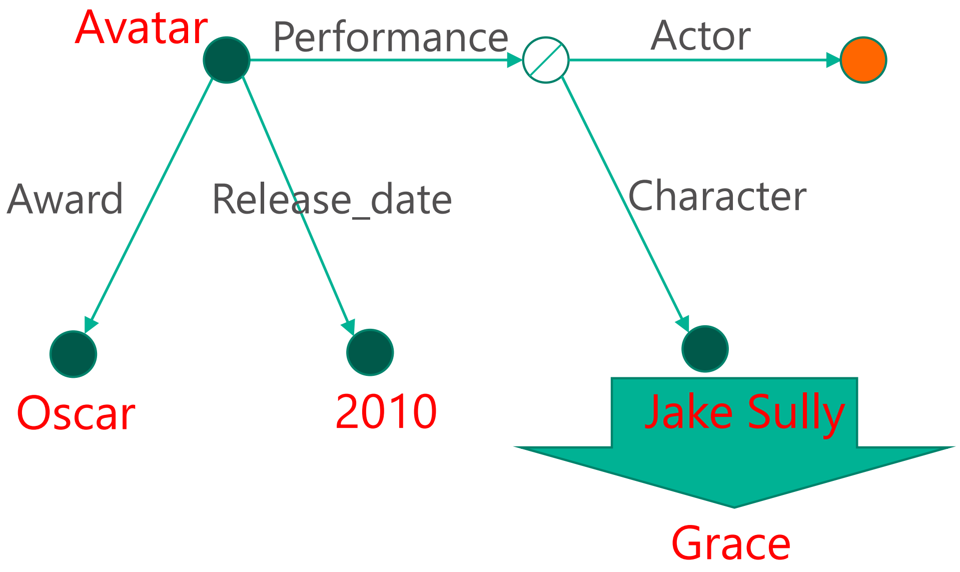
- Nodes whose values were observed
- Nodes questioned in user's last utterance
- Internal Nodes

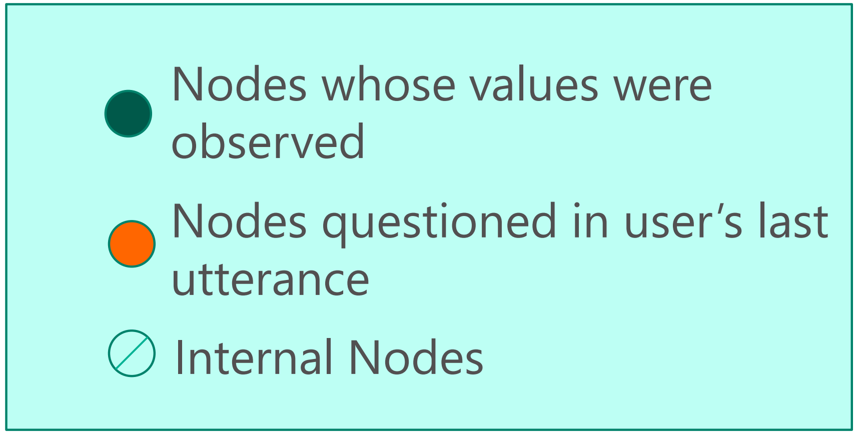
Simplified SPARQL Query:

```
SELECT ?actor WHERE {
        ?movie name "Avatar".
        ?movie performance ?perf.
        ?perf actor ?actor.
        ?perf character "Jake Sully".
        ?movie release_date "2010".
        ?movie award "Oscar".
}
```

Example 1

U1: Who played Jake Sully in the Oscar winning 2010 movie Avatar?
U2: How about Grace?





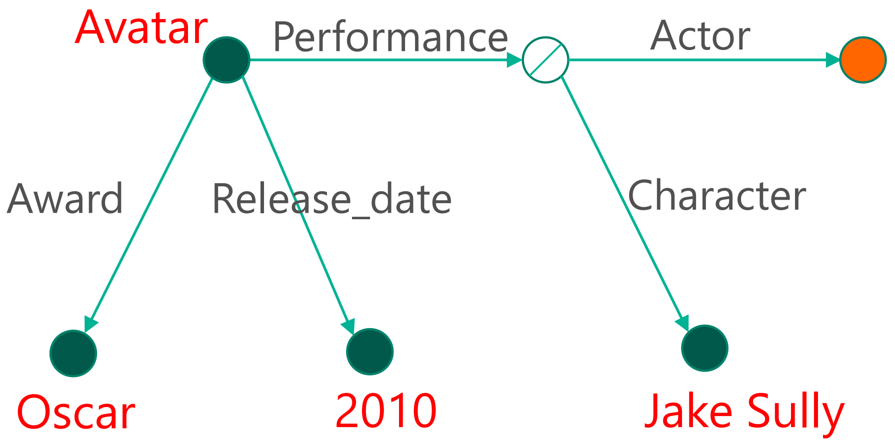
Override slot value

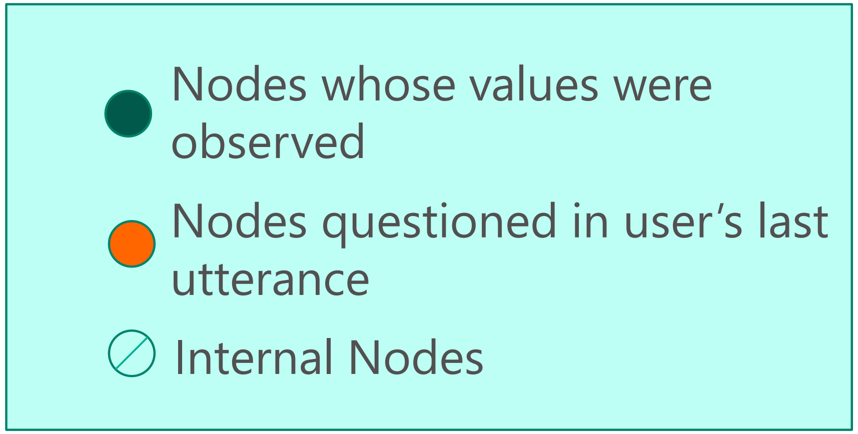
Simplified SPARQL Query:

```
SELECT ?actor WHERE {
        ?movie name "Avatar".
        ?movie performance ?perf.
        ?perf actor ?actor.
        ?perf character "Grace".
        ?movie release_date "2010".
        ?movie award "Oscar".
}
```

Example 1

U1: Who played Jake Sully in the Oscar winning 2010 movie Avatar?
U2: I wanna change my xbox avatar

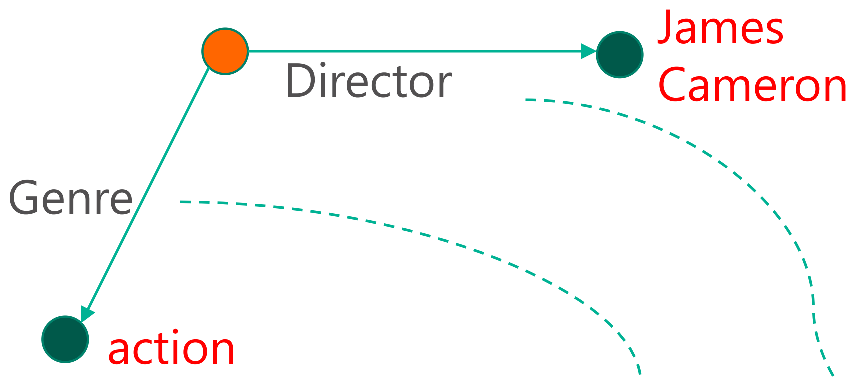




Domain Switch

Example 2

U1: Find some action movies by James Cameron



Nodes whose values were observed

Nodes questioned in user's last utterance

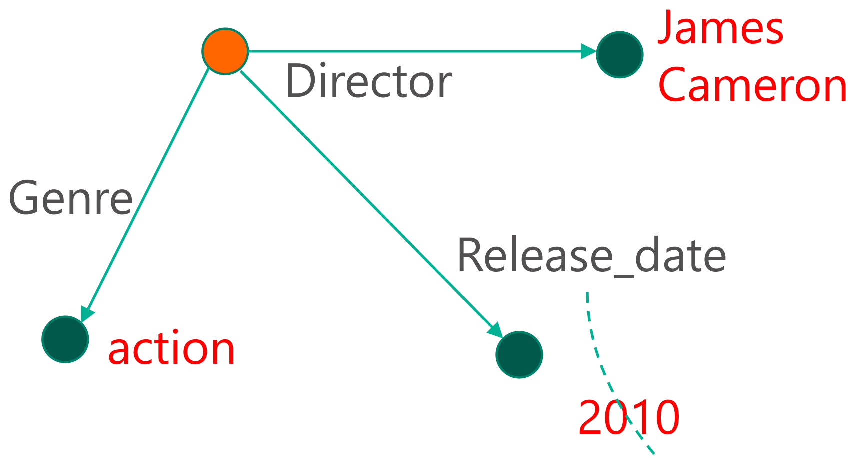
Simplified SPARQL Query:

```
SELECT ?movie WHERE {
    ?movie director "James Cameron".
    ?movie genre "Action".
}
```

Example 2

U1: Find some action movies by James Cameron

U2: Which of these were released in 2010?



- Nodes whose values were observed
- Nodes questioned in user's last utterance

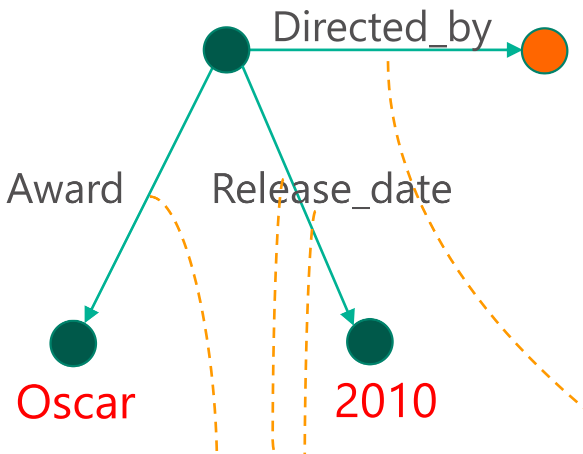
Add another constraint

Simplified SPARQL Query:

```
SELECT ?movie WHERE {
        ?movie director "James Cameron".
        ?movie genre "Action".
        ?movie release_date "2010".
}
```

Example 3

U1: Who directed the Oscar winning 2010 movie Avatar?



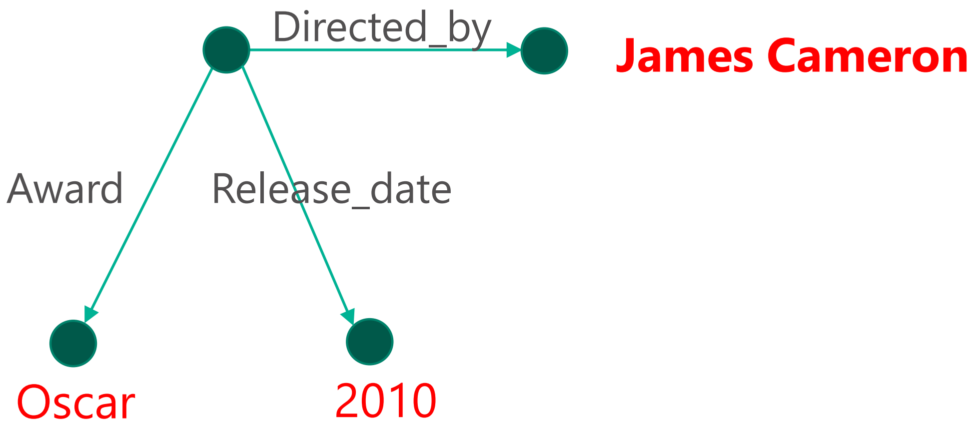
- Nodes whose values were observed
- Nodes questioned in user's last utterance

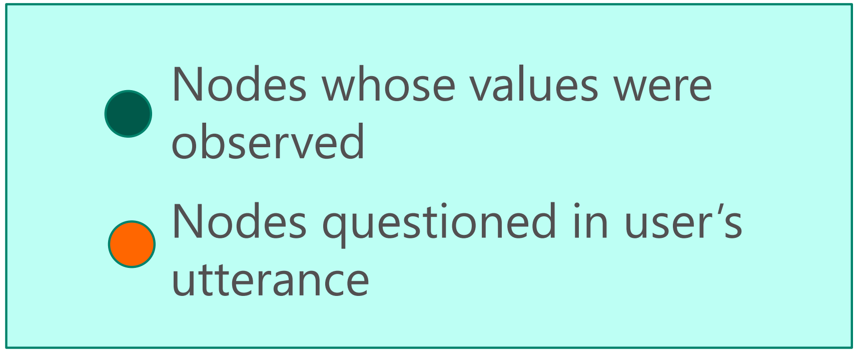
Simplified SPARQL Query:

```
SELECT ?actor WHERE {
        ?movie name "Avatar".
        ?movie directed_by ?director.
        ?movie release_date "2010".
        ?movie award "Oscar".
}
```

Example 3

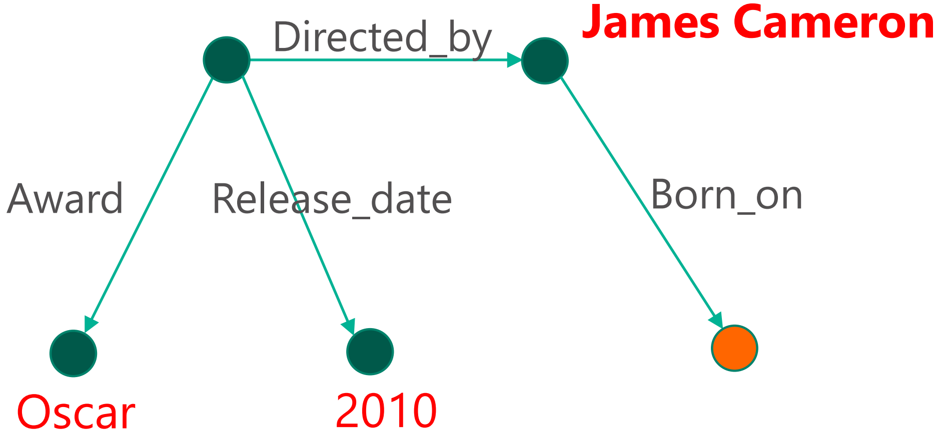
U1: Who directed the Oscar winning 2010 movie Avatar?

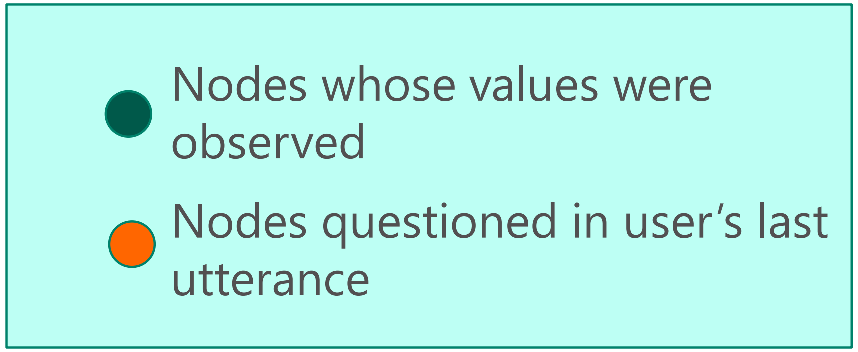




Example 3

U1: Who directed the Oscar winning 2010 movie Avatar?
U2: When was he born?

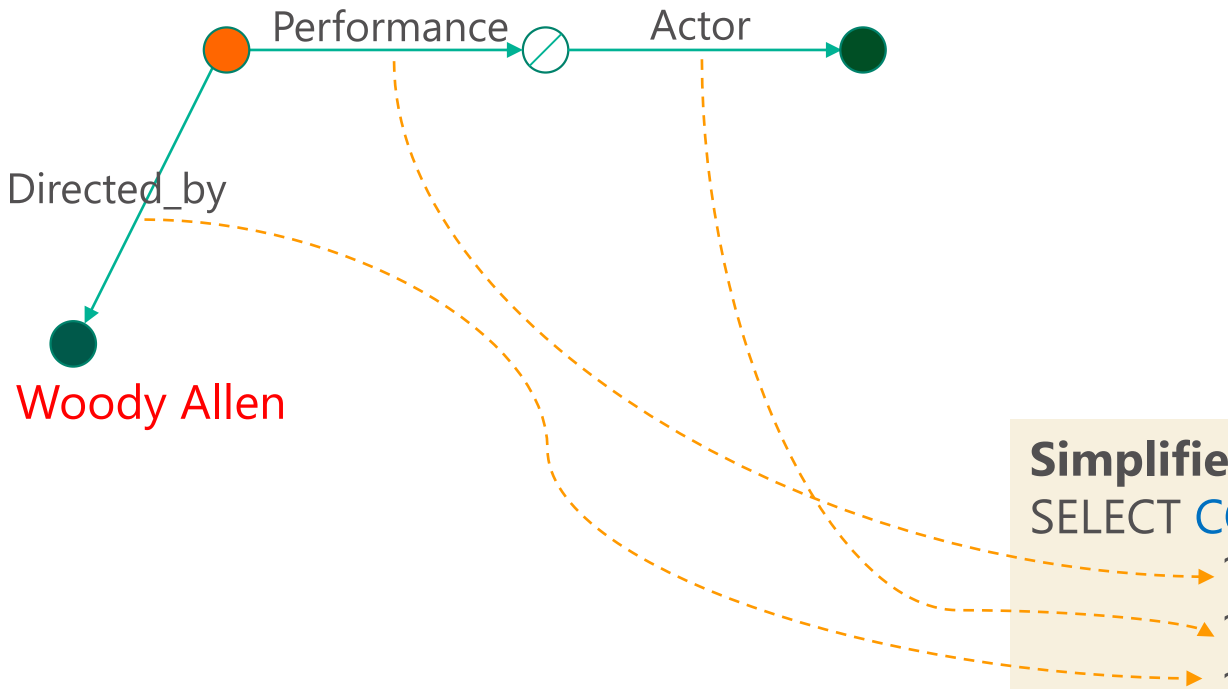




Trace the path!

Example 4

U1: How many Woody Allen movies star Diane Keaton?



- Nodes whose values were observed
- Nodes questioned in user's last utterance
- Internal Nodes

Simplified SPARQL Query:

```
SELECT COUNT(?movie) AS ?count WHERE {
    ?movie performance ?perf.
    ?perf actor ?actor.
    ?movie directed_by "Woody Allen".
}
```

Outline

- Conversational systems
- Sematic Web and Linked Data Sources
- **SLU based on knowledge graphs**
- Leveraging Context with knowledge graphs
- Learning from the Semantic Web

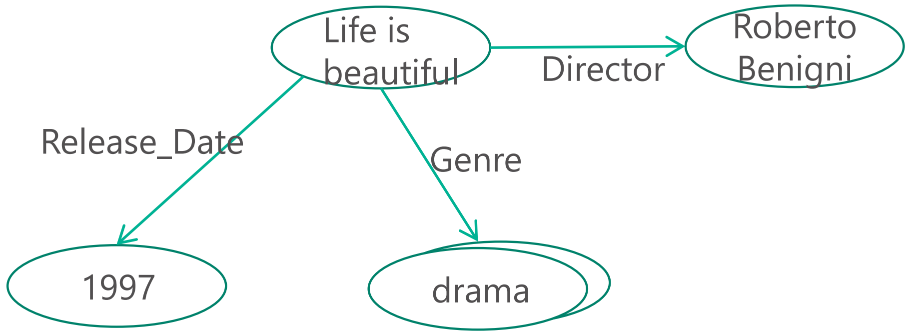
SLU based on Knowledge Graphs

- Semantic Representation: Schema of knowledge graphs
- Represent user utterances as a sub-graph of entities and **relations.**

- Pros:
  - The semantic space is **already** defined by domain experts.
  - Can take advantage of the knowledge graph as prior knowledge.
  - Can take advantage of other “aligned” resources, such as Wikipedia or query click logs.
  - Interpretation is simpler, as there is a direct mapping to query languages.

SLU based on Knowledge Graphs

Sample from the relevant part of the knowledge graph:



Sample user utterances:	*Show me movies by Roberto Benigni*	*Who directed Life is Beautiful?*
Corresponding relation on the knowledge graph		

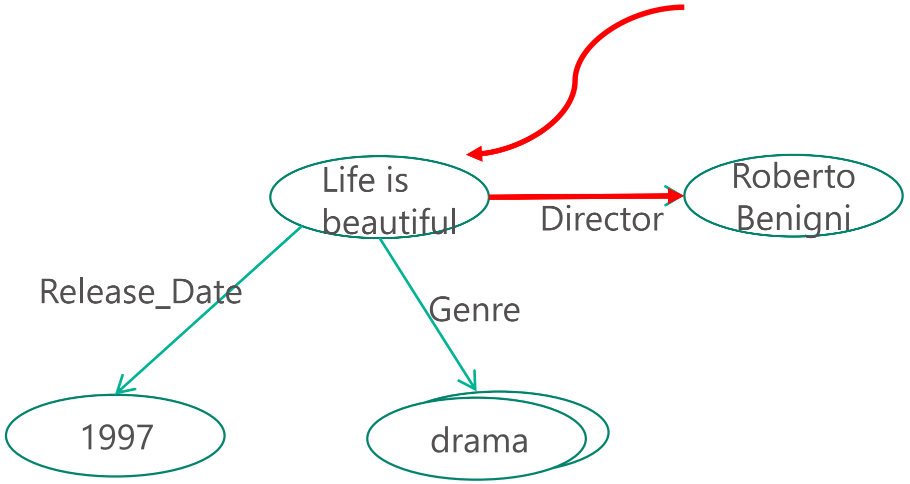
Request in SPARQL (simplified for demonstration):	SELECT ?*movie* WHERE
	{
	?*movie* Director "Roberto Benigni".
	}
		SELECT ?*director* WHERE
		{
		"Life is Beautiful" Director ?*director*.
		}

Request in logical form (simplified):	$\lambda y.\exists x.x=\text{"Roberto Benigni"} \wedge \text{Director}(x,y)$	$\lambda x.\exists y.y=\text{"Life is beautiful"} \wedge \text{Director}(x,y)$

SLU based on Knowledge Graphs - Overview

- Entity Extraction
- Entity Linking
- Relation Detection

Who directed Life is Beautiful?



Slots/Intents versus Relations

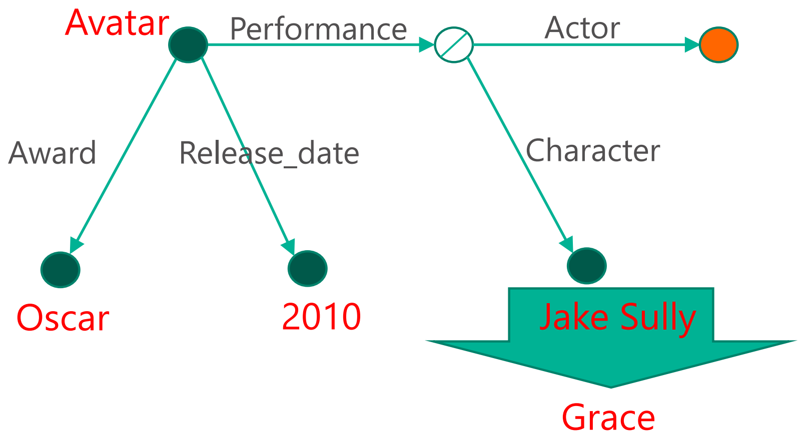
- Relations combine intents and slots:
  - Direct mapping to knowledge graphs allows for data mining
  - Easily confusable intents are merged:

Example user utterances:	Intents	Slots	relations
Who directed Avatar	Find_director	Movie_name	Director, mname
Show movies James Cameron directed	Find_movie	Director_name	Director, dname
Find me some action movies	Find_movie	Genre	genre
Did James Cameron direct Avatar	Find_movie/ Find_director?	Director_name, Movie_name	Director,dname, mname

- Intents are designed according to dialog acts and expected result:
  - Find_movie class is overly populated (more than 70% of the data)
- Joint modeling of intents and slots comes naturally with relations.

Following Turns: Information State Updates

- *Information State*: Information necessary to distinguish a dialog from others, representing cumulative additions from previous actions in the dialog [Traum & Larsson, 2003].
- Key: Update of information states as the dialog progresses.



Following Turns: Information State Updates

- Our approach:
  - Updates: Domain switch, override slot, add new constraint,…
  - Focuses on multi-domain systems.
  - Uses domain and task independent features, hence it is possible to introduce new domains seamlessly.
  - Simple and flexible: One classification model for all updates.
- The update actions can be used as is to determine the next state, or as input feature to reduce the state space.

More details in:

- Dilek Hakkani-Tur, Gokhan Tur, Larry Heck, Ashley Fidler, and Asli Celikyilmaz, A Discriminative Classification-Based Approach to Information State Updates for a Multi-Domain Dialog System, Proc. Interspeech, September 2012

Related Work on Relation Detection and Knowledge Graphs

- Relation Extraction
- Distant learning from knowledge graphs
- Large scale semantic parsing and question answering

Relation Extraction

- Studied in information extraction.
- **Aim:** finding instances of specific relations in documents, between pairs of entities, i.e.:
  - Person-affiliation
    - John Smith is the Chief Scientist of the Hardcom Corporation
      (example from Zelenko et al., 2003)
  - Organization-location
- Classification of sequences/parses, using syntax:
  - Kernel methods [ZelenkoEtAl, 2003]
  - Dependency tree kernels [Culotta&Sorensen, 2004]
  - Shortest path tree kernels [Bunescu&Mooney, 2005]

Distant Learning

- (Wu & Weld, 2007)
- Semantic Web perspective: method of automatically structuring a large amount of existing data.
- Transferred labels from existing Wikipedia infoboxes to corresponding documents.
- Used the automatically labeled data for training extractors.
- Use extractors to form new/complete infoboxes.

California

From Wikipedia, the free encyclopedia

For other uses, see California (disambiguation).

California (/ˌkælɨˈfɔrnjə/) is a state located on the West Coast of the United States. It is the most populous U.S. state,[12] home to one out of eight people who live in the U.S., with a total of 38 million people, and it is the third largest state by area (after Alaska and Texas). California is bordered by Oregon to the north, Nevada to the east, Arizona to the southeast, and the Mexican State of Baja California to the south. It is home to the nation's second and fifth most populous census statistical areas (Greater Los Angeles Area and San Francisco Bay Area, respectively), and eight of the nation's 50 most populated cities (Los Angeles, San Diego, San Jose, San Francisco, Fresno, Sacramento, Long Beach, and Oakland).[13] Sacramento is the state capital.

State of California

Flag

Seal

Nickname(s): *The Golden State*

Motto(s): *Eureka*[1]

Official language	English

Spoken languages	As of 2007
	• English 57.4%[2]
	• Multilingual 32.8%[2]
	  • Spanish 28.5%[3]
	  • Chinese 2.8%[3]
	  • Tagalog 2.2%[3]
	  • Vietnamese 1.43%[3]

Demonym	Californian
Capital	Sacramento
Largest city	Los Angeles
Largest metro	Greater Los Angeles Area

Sample Wikipedia Infobox

Distant Learning

- (Mintz et al., 2009)
  - Freebase-scale relation detection.

Relation name	Size	Example
/people/person/nationality	281,107	John Dugard, South Africa
/location/location/contains	253,223	Belgium, Nijlen
/people/person/profession	208,888	Dusa McDuff, Mathematician
/people/person/place_of_birth	105,799	Edwin Hubble, Marshfield
/dining/restaurant/cuisine	86,213	MacAyo's Mexican Kitchen, Mexican

  - Search for Wikipedia sentences that contain related entity pairs.

[Steven Spielberg]'s film [Saving Private Ryan] is loosely based on the brothers' story.

Allison co-produced the Academy Award-winning [Saving Private Ryan], directed by [Steven Spielberg]...

  - Use the matching sentences to train relation extractors
  - Features that describe how two entities relate to each other in a sentence:
    - Lexical features, i.e., sequence of words between two entities, or their POS tags.
    - Syntactic features, i.e., dependency path between two entities.

Semantic Parsing and Question Answering

- (Cai & Yates, 2013)
  - Semantic parser for Freebase used for parsing questions (data set from 81 Freebase domains).
  - Schema matching to identify phrases that correspond with each relation & entity:
    - Search for entity pairs & identify patterns for each relation.
  - Extend a semantic parser based on Probabilistic Combinatory Categorial Grammar (PCCG) with schema matching.
- (Berant et al., 2013)
  - A logical language: Lambda Dependency-Based Compositional Semantics(λ-DCS)
  - 15 mil. triples (e_1; r; e_2) from ClueWeb09 using ReVerb System (Fader et al., 2011)

    ("Obama", "was also born in", "August 1961")
- (Yao & vanDurme, 2014)

Outline

- Conversational systems
- Sematic Web and Linked Data Sources
- SLU based on knowledge graphs
- **Leveraging Context with knowledge graphs**
- Learning from the Semantic Web

Leveraging Context with Knowledge Graphs

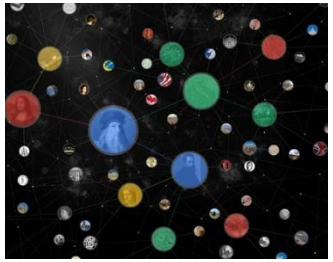
- Visual
  - Passive: content displayed to user provides context for conversation and anchor points (entities) to KG
  - Active: **multi-modal** interaction with KG entity(ies) through display
- Dialog

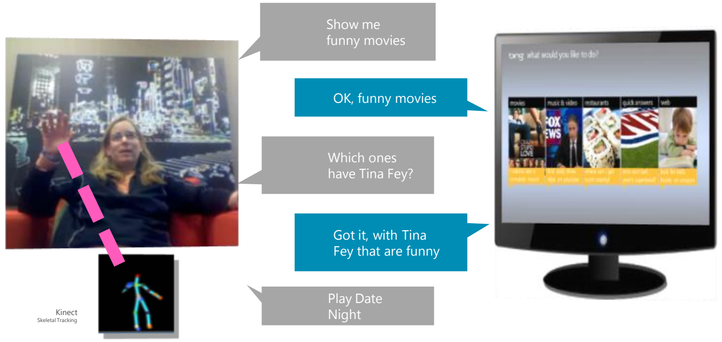
  Coherence (topic, speaker, etc.) of multi-turn conversations
- Personal

  Personal KG (spouse, profession, etc.), historical interactions, preferences
- Location (Geo)

  Current location influences conversational interactions
- Time/Day/Season

  Conversational topics related to the moment in time





Personal Assistant for Phones

Send e-mail to Larry about the demo

Wake me up tomorrow at 6

Other Screens:

Look for Portlandia clips on hulu

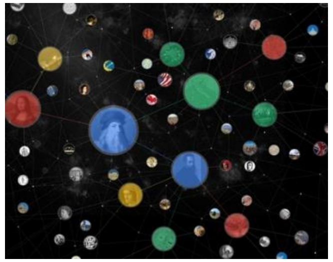
How about places with outdoor dining?

Leveraging Context with Knowledge Graphs

- **Visual**
  - Passive: content displayed to user provides context for conversation and anchor points (entities) to KG
  - Active: **multi-modal** interaction with KG entity(ies) through display
- Dialog

  Coherence (topic, speaker, etc.) of multi-turn conversations
- Personal

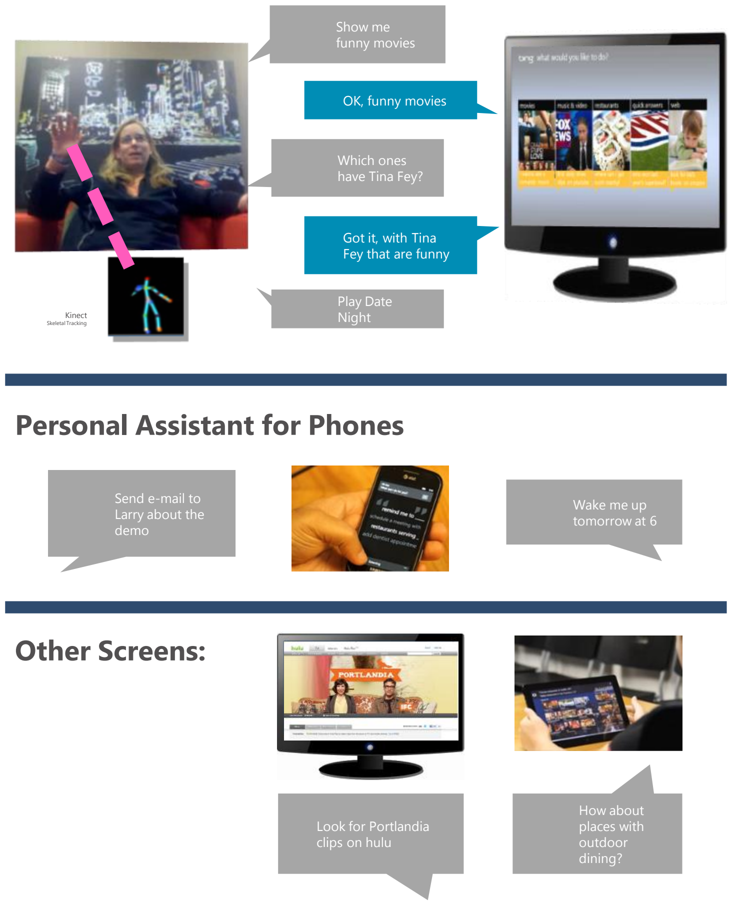
  Personal KG (spouse, profession, etc.), historical interactions, preferences



- Location (Geo)

  Current location influences conversational interactions
- Time/Day/Season

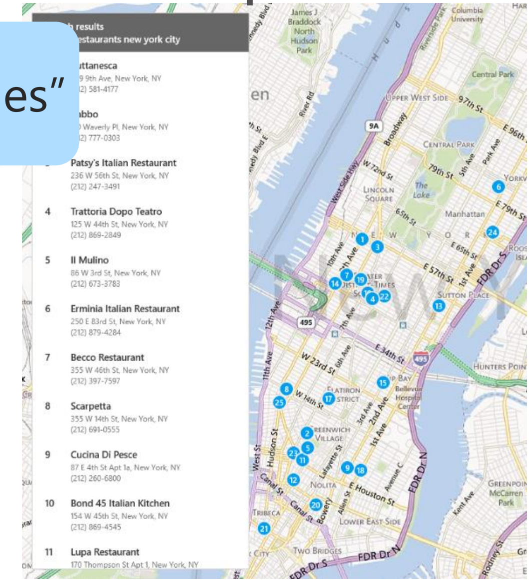
  Conversational topics related to the moment in time



Leveraging Context with Knowledge Graphs

Visual Context

“family friendly ones”



Processing Flow of an Interaction with Knowledge

- Scoping

    Linking the focus entity(ies) to the knowledge graph

    E.g., circling map and speaking

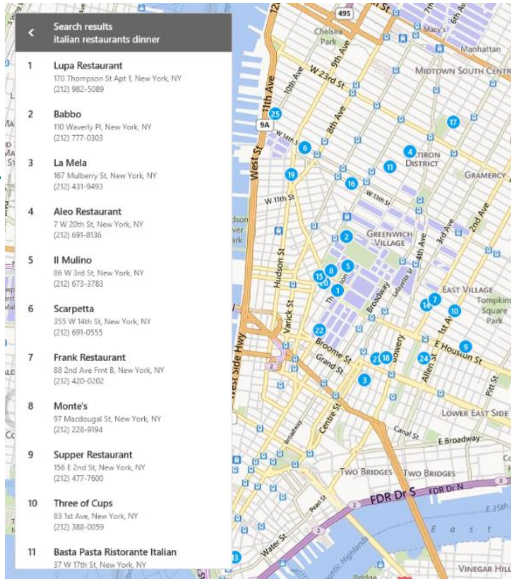
- Intent detection

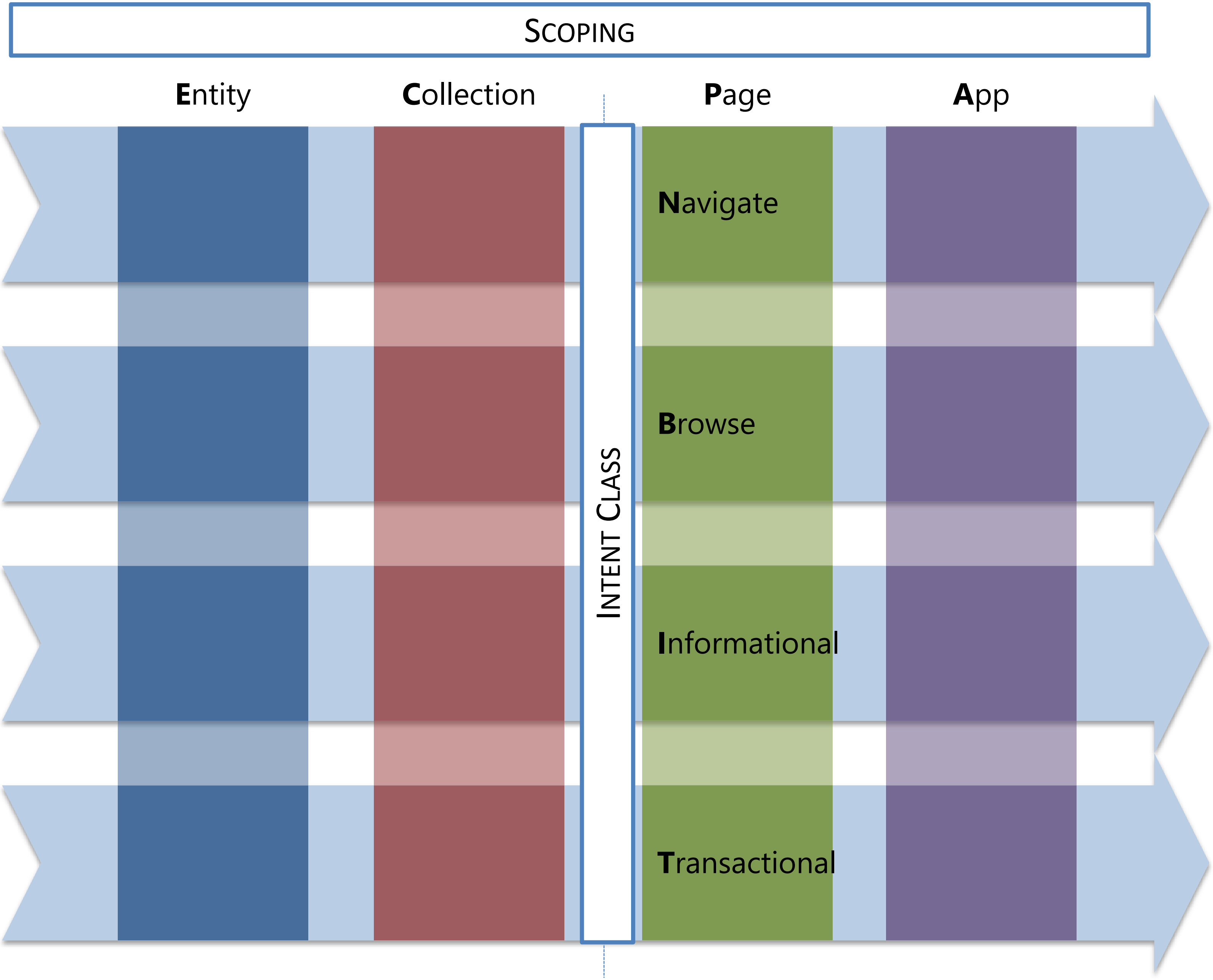
    Interpret the intent (what the user wants) in
    the context of the knowledge (and other context)

- Execution

    Send request to an appropriate execution engine

Browse





Taxonomy of the (Conversational Knowledge) Web

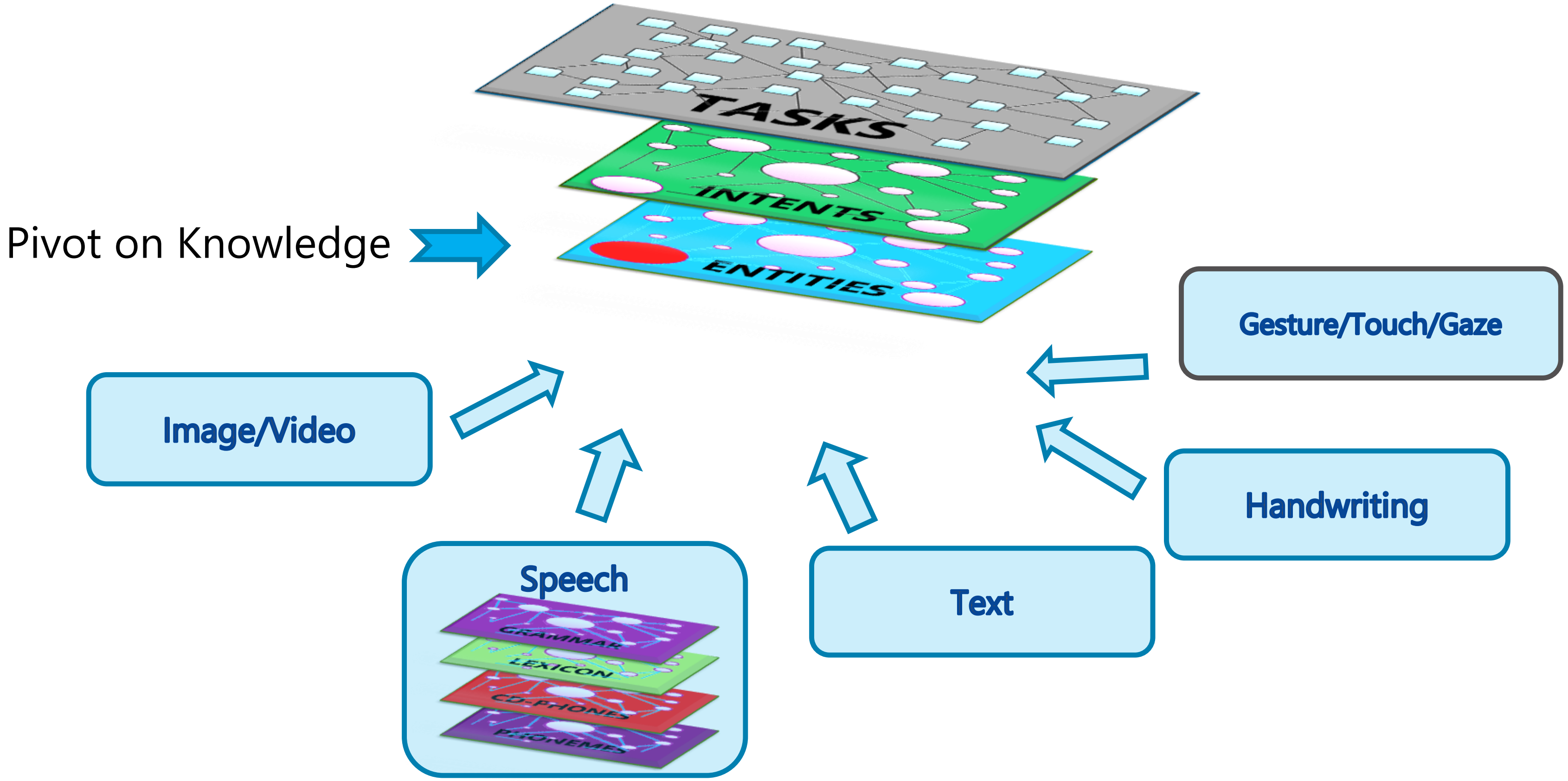
Taxonomy of the (Conversational Knowledge) Web

SCOPING

INTENT CLASS	Entity	Collection	Page	App
Navigate	Navigate to entity homepage	---	Navigate hyperlinks	Navigate menu/options
Browse	Recommend related entities/content (more like this)	Filter/Expand collection	Recommend content (more like this)	Recommend apps (more like this)
Informational	Factoid and List questions	Comparative questions (e.g., best, top-k)	Page domain questions. (e.g., “Will it rain today?”)	App domain questions. (e.g., “Will it rain today?”)
Transactional	Perform action on entity (e.g., book a table for two)	Perform action on a collection (e.g., share, compare)	Generic actions such as “share page”, or explicit page actions	Deep links to app functionalities

Leveraging Context with Knowledge Graphs

Visual Context: "Scoping" Extends to Other Modalities



Leveraging Context with Knowledge Graphs

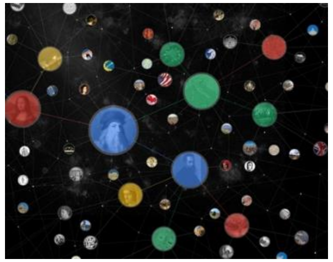
- Visual
  - Passive: content displayed to user provides context for conversation and anchor points (entities) to KG
  - Active: **multi-modal** interaction with KG entity(ies) through display
- **Dialog**

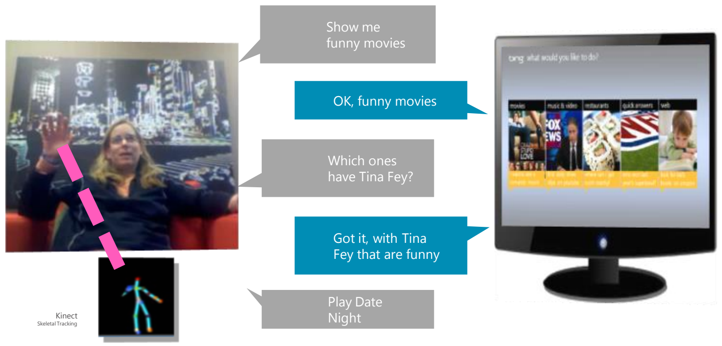
  Coherence (topic, speaker, etc.) of multi-turn conversations
- Personal

  Personal KG (spouse, profession, etc.), historical interactions, preferences
- Location (Geo)

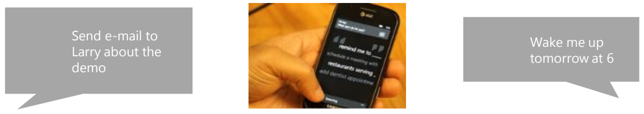
  Current location influences conversational interactions
- Time/Day/Season

  Conversational topics related to the moment in time

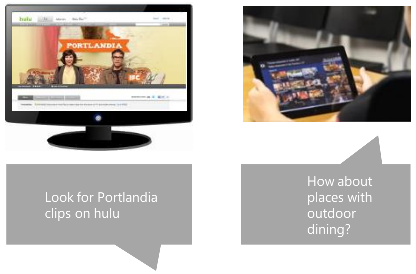




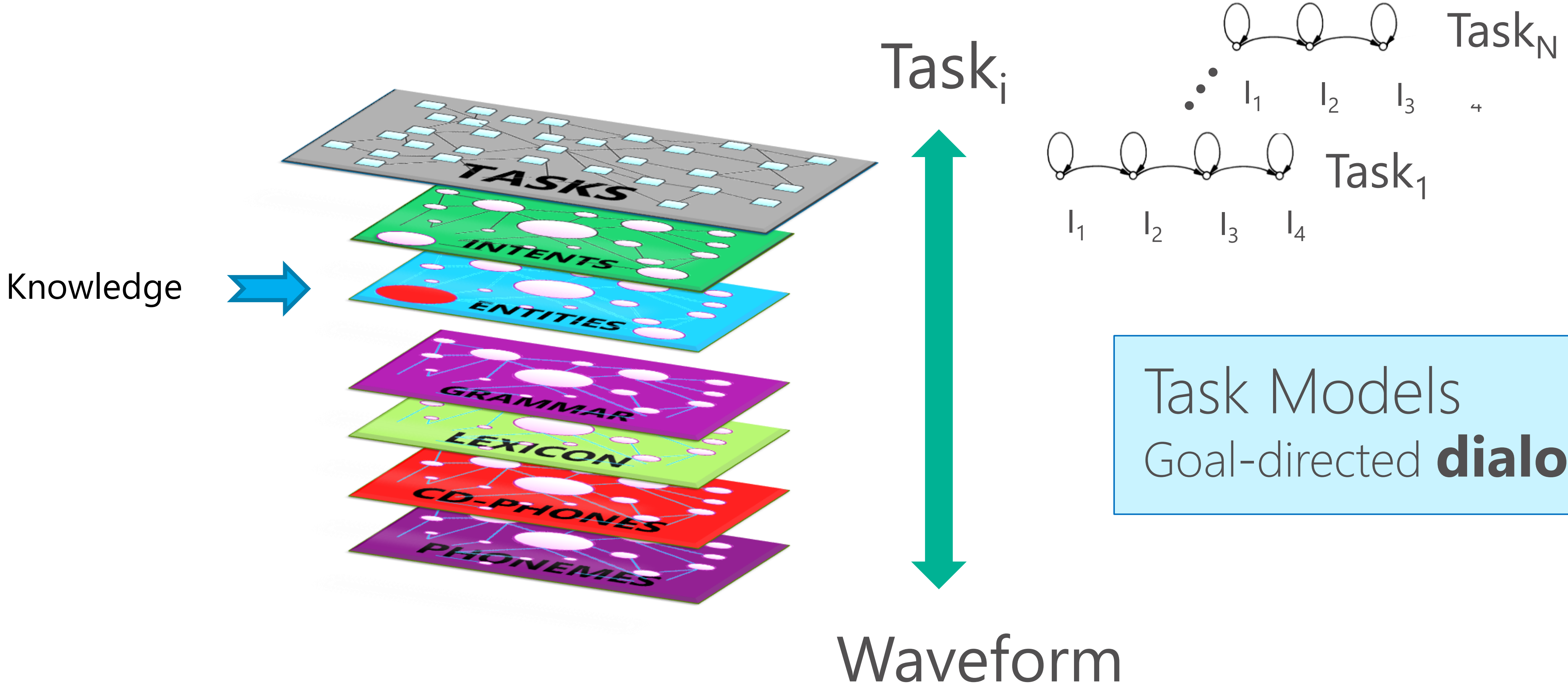
Personal Assistant for Phones



Other Screens:



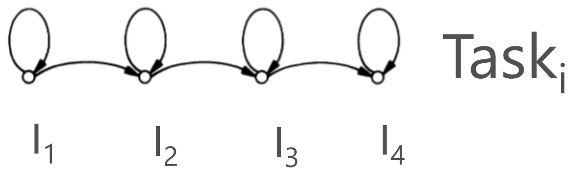
Leveraging Context with Knowledge Graphs
Dialog Context



Task Models
Goal-directed **dialogs**

Leveraging Context with Knowledge Graphs

Dialog Context: KG-based Dialog Modeling



Statistical methods for dialog managers is active area (e.g., POMDP)

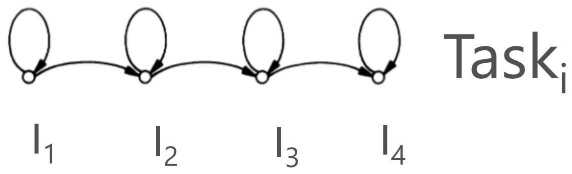
Key Technical Challenge: amount of annotated dialogs required for training

Idea: leverage Web (IE) session data combined with Knowledge Graphs

Web search & browse ➔ Conversations/dialog
Massive volume of interactions >100M queries/day, Millions of users
Coverage of user interactions is high (broad domains across the web)

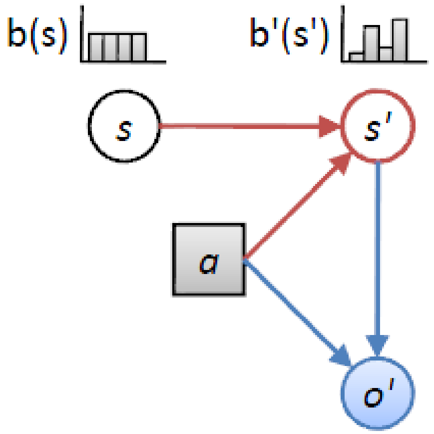
Leveraging Context with Knowledge Graphs

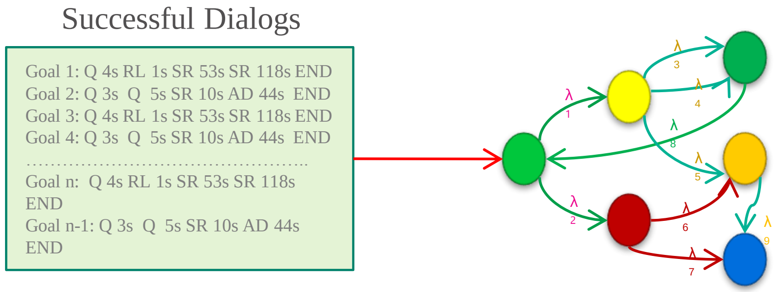
Dialog Context: KG-based Dialog Modeling

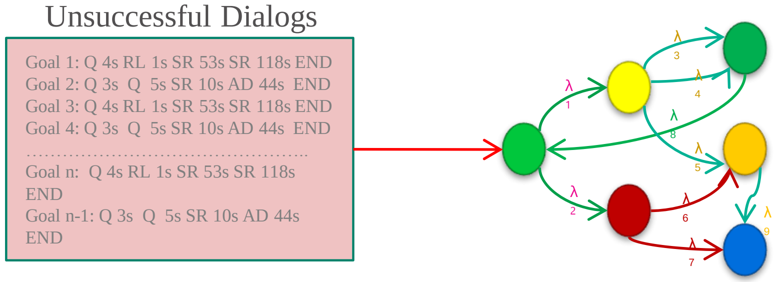


New Approach

Step 1. ***Learn task completion patterns from web*** → IE sessions through Satori KG



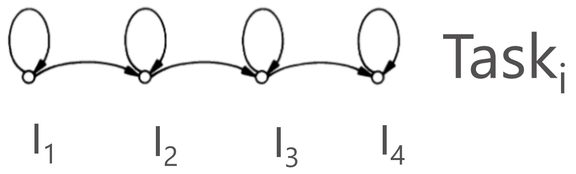




Step 2. ***Bootstrap*** *multi-turn spoken dialog models from Knowledge Graph traversal paths*

Leveraging Context with Knowledge Graphs

Dialog Context: KG-based Dialog Modeling



Results

Successfully learned ***conversational search and browse*** models from **IE sessions** + **Satori**

Increased F-measures of semantic parsing by **> 18% (rel.)**

1st method to directly leverage **web browse patterns** & **knowledge graphs** to **bootstrap spoken dialog models**

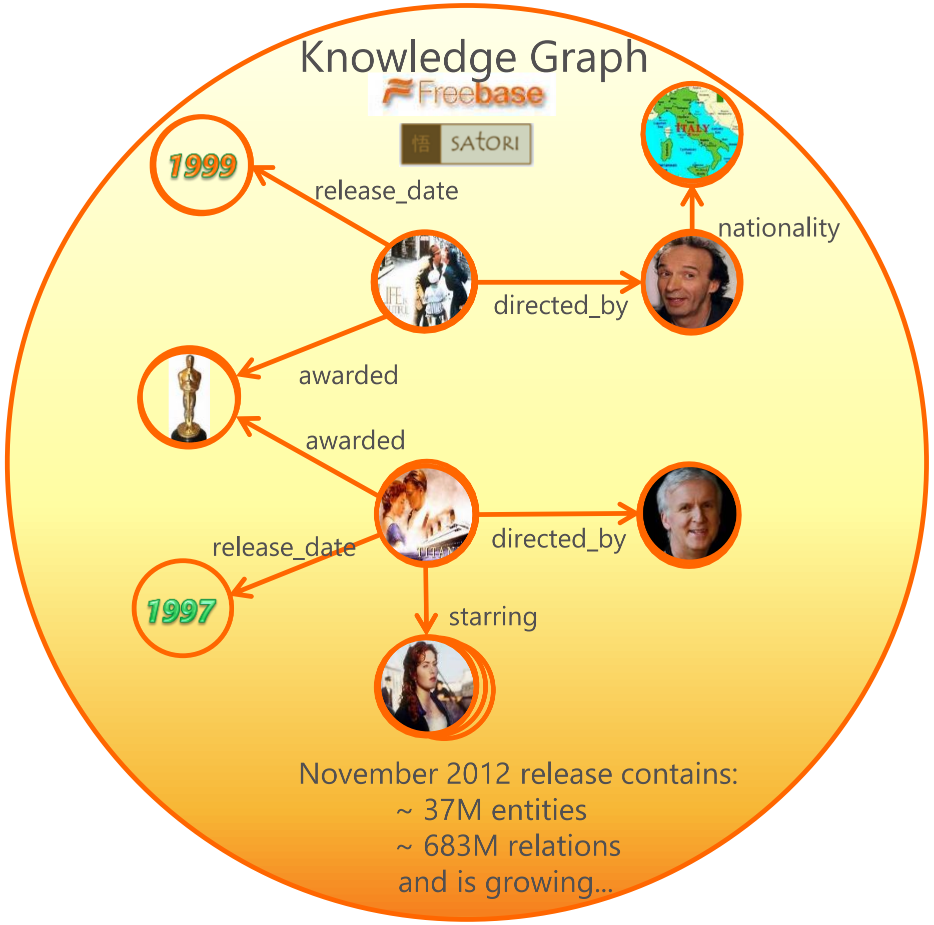
To dig deeper…

Lu Wang, Larry Heck, Dilek Hakkani-Tur, Leveraging Semantic Web Search and Browse Sessions for Multi-Turn Spoken Dialog Systems, *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2014*

Outline

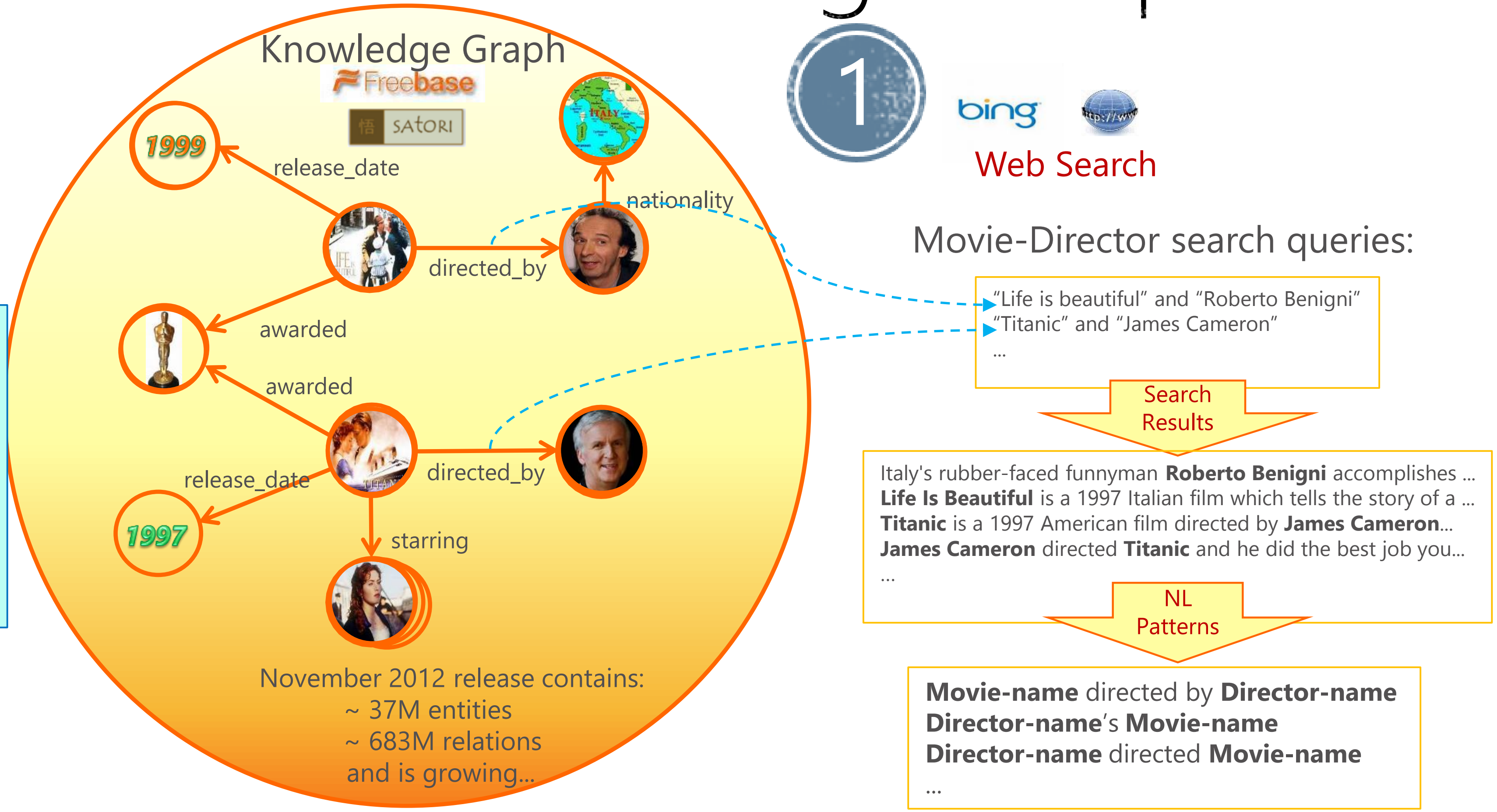
- Conversational systems
- Sematic Web and Linked Data Sources
- SLU based on knowledge graphs
- Leveraging Context with knowledge graphs
- **Learning from the Semantic Web**
  - Learning Entity Extraction and Relation Detection from Data Linked to Knowledge Graphs
    Examples for learning the mappings from utterances to relations and entities
  - New Intent Discovery
    Relations or transactions missing on the graph
  - Introducing Weights to the Knowledge Graph (special case: entity types)
  - Deep Learning from Knowledge Graphs

Linking Data to the Knowledge Graph



Linking Data to the Knowledge Graph

- Search for all pairs of entities connected by the same relation [Ravichandran& Hovy, 2002]
- Extract relation patterns from these



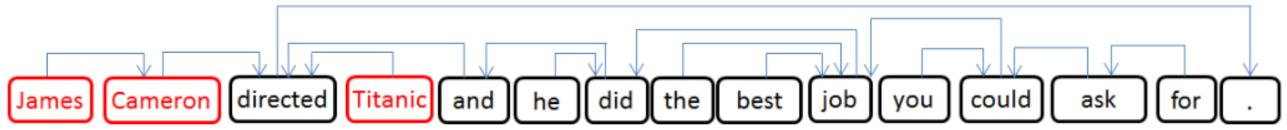
Extracting Relation Patterns

- Filter search results

  S_{ab}: set of all snippets returned by search

  $$M_{ab} = \{s : s \in S_{ab} \wedge includes(s,a) \wedge includes(s,b)\}$$

- Use dependency parsing to find the right span.
  - The smallest sub-tree that includes both entities
- Replace entity names with entity type tokens.



James Cameron directed Titanic

Director-name directed Movie-name

Other examples:

Microsoft has 130,000 employees that are …

Avatar is an action movie directed by …

Extracting Relation Patterns

- Find salient and discriminating patterns using mutual information, p_j denotes a pattern and r_i denotes a relation:

$$I(p_j, r_i) = \log P(r_i|p_j)/P(r_i)$$

And KL divergence:

$$KL(P(r)||P(r|p))$$

Director-name and Movie-name ✗
Director-name directed Movie-name ✓
Movie-name is directed by Director-name ✓
Actor-name and Movie-name ✗
Movie-name in Movie-location ✗
Movie-name in Movie-release-date ✗
Movie-name was released in Movie-release-date ✓

More details in:

- Dilek Hakkani-Tur, Larry Heck, and Gokhan Tur, Using a Knowledge Graph and Query Click Logs for Unsupervised Learning of Relation Detection, IEEE ICASSP, May 2013.

Relation Detection Models

- Multi-label, multi-class classification models for relation detection.
- Sequence models for slot filling/entity detection.

- Select patterns that discriminate different relations.
- To tackle label transfer errors, iterate model training (via self learning).

- Continuously adapt models with the collected data.

Relation Detection Experiments

- AdaBoost, with word {1,2,3}-grams.
- Movies domain, 7 entity pairs: movie-director, -star, -release_date, -MPAA-rating, -nationality, -language, -genre
- **Mined data:** 178K patterns with Bing search.
- **Development and test sets:**
  - 1200 user utterances each, manually labeled with 20 relations.
  - 66% of the development and 64% of the test set examples include one of the 7 relations.
- **Supervised training set:**
  - 2334 examples of 7 relations collected via crowdsourcing, and annotated manually.

Results

	Micro-F	Targeted Macro-F
Majority Class	20.3%	4.2%
Full Snippets	42.5%	55.1%
Patterns from Snippets	44.1%	58.0%
Patterns from Snippets (1 iter)	45.2%	59.6%
Search Queries	31.6%	40.6%
Search Queries (1 iter)	34.7%	43.2%
Combination (upper bound)	50.2%	62.7%
Combination (weighted voting)	**45.5%**	**59.9%**
Supervised	47.6%	59.3%

Evaluation:

Micro-F: F-measure for all 20 categories (remaining 13 categories are not captured with unsupervised methods)

Targeted Macro-F: Macro-averaged F-measure for 7 relations.

Ongoing/Future Work

- Changing the modeling of relation detection:
  - Tagging pairs of slots
  - Tagging syntactic dependencies
- Modeling as a parsing or optimization problem
  - Tradeoff: performance gain vs. complexity
- Checking wellness of formed queries using the graph
- Modeling the second turn
  - Would we gain from using the knowledge in the graph for information state updates

Linking Data to the Knowledge Graph

2

Wikipedia
& other document sources

Article | Talk | Read | Edit | Vi

Life Is Beautiful

From Wikipedia, the free encyclopedia
(Redirected from Life is beautiful)

For other uses, see Life Is Beautiful (disambiguation).

Life Is Beautiful (Italian: *La vita è bella*) is a 1997 Italian comedy-drama film directed by and starring Roberto Benigni. Benigni plays Guido Orefice, a Jewish Italian book shop owner, who must employ his fertile imagination to shield his son from the horrors of internment in a Nazi concentration camp. Part of the film came from Benigni's own family history; before Roberto's birth, his father had survived three years of internment at the Bergen-Belsen concentration camp. The film was a critical and financial success, winning Benigni the Academy Award for Best Actor at the 71st Academy Awards as well as the Academy Award for Best Original Dramatic Score and the Academy Award for Best Foreign Language Film.

Contents [hide]
1 Plot

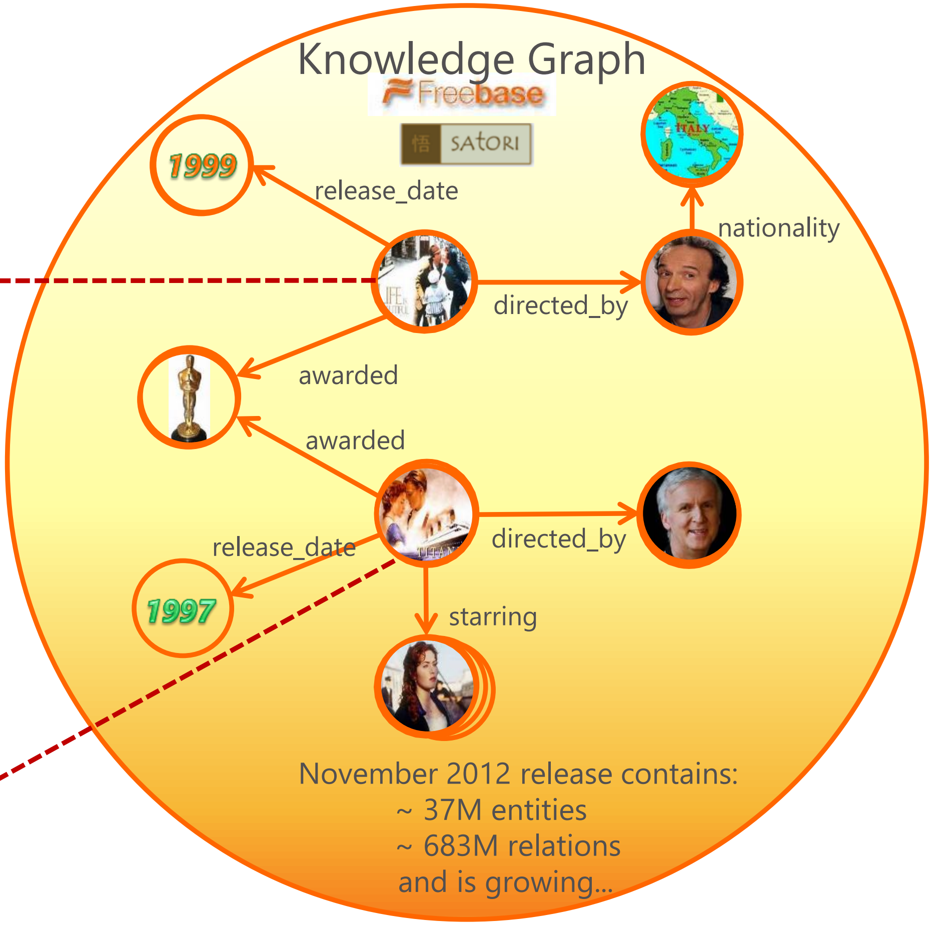
Article | Talk | Read | View source | Vi

Titanic (1997 film)

From Wikipedia, the free encyclopedia

Titanic is a 1997 American epic romantic disaster film directed, written, co-produced, and co-edited by James Cameron. A fictionalized account of the sinking of the RMS *Titanic*, it stars Leonardo DiCaprio and Kate Winslet as members of different social classes who fall in love aboard the ship during its ill-fated maiden voyage.

Cameron's inspiration for the film was predicated on his fascination with shipwrecks; he wanted to convey the emotional message of the tragedy, and felt that a love story interspersed with the human loss would be essential to achieving this. Production on the film began in 1995, when Cameron shot footage of the actual *Titanic* wreck. The modern scenes were shot on board the *Akademik Mstislav Keldysh*, which Cameron had used as a base when filming the wreck. A reconstruction of the *Titanic* was built at Playas de Rosarito, Baja California, and scale models and computer-generated imagery were also used to recreate the sinking. The film was partially funded by Paramount Pictures and 20th Century Fox, and, at the time, was the most expensive film ever made, with an estimated budget of $200 million.



- Transfer labels from the knowledge graph to documents

More details in:

- Larry Heck, Dilek Hakkani-Tur, and Gokhan Tur, Leveraging Knowledge Graphs for Web-Scale Unsupervised Semantic Parsing, *Proc. Interspeech*, August 2013.

Leveraging KGs for Semantic Parsing

Procedure

Unsupervised Data Mining with Knowledge Graphs

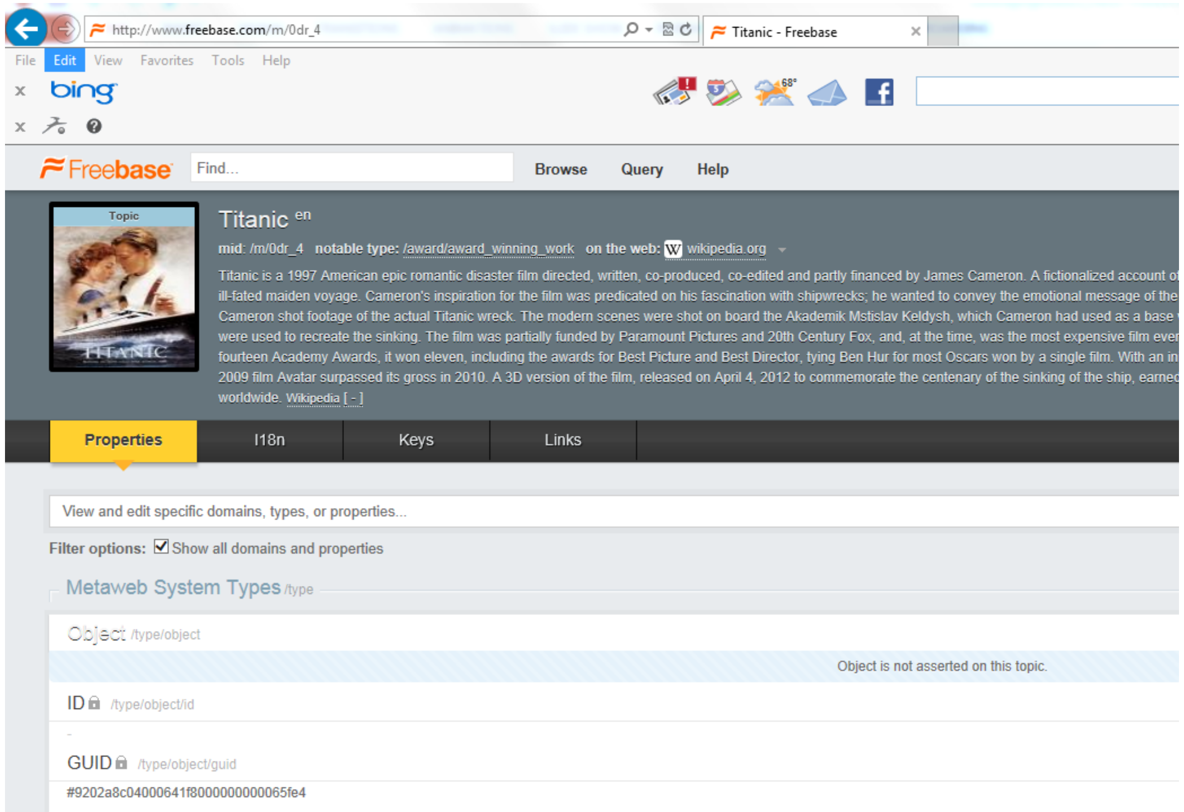
- 6 step procedure
- Auto-annotated (unsupervised) data used to train SLU

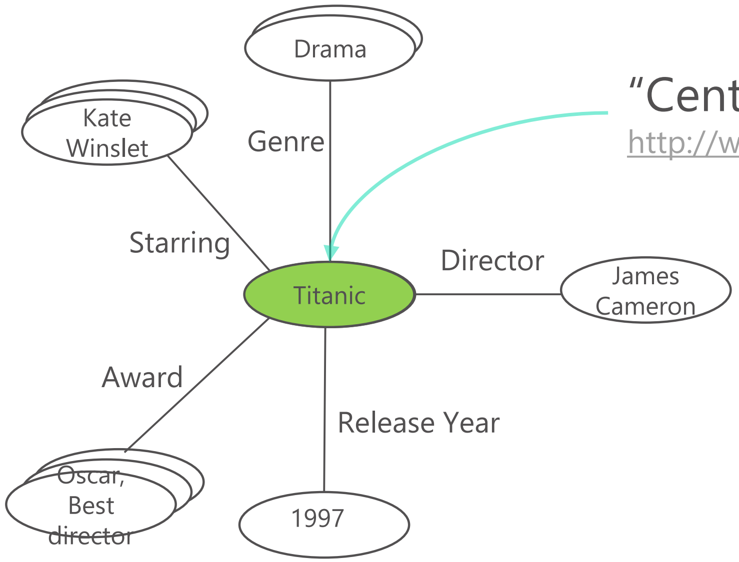
Style Adaptation

Modeling Relations for Semantic Parsing

Unsupervised Data Mining with KGs

Step #1: Select starting node in graph (entity type)





"Central Pivot Node (CPN)"

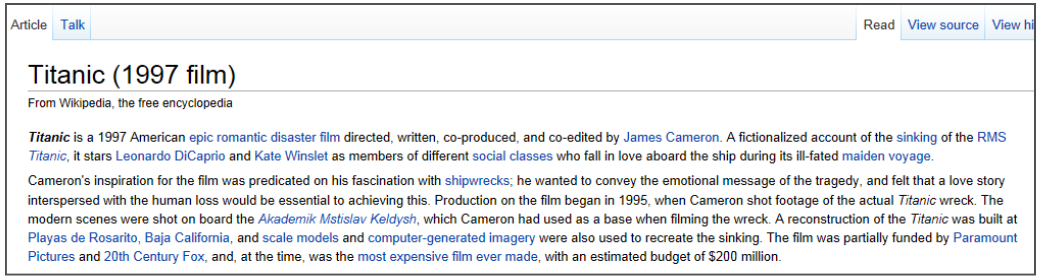
http://www.freebase.com/m/0dr_4

Unsupervised Data Mining with KGs

Step #2: Get Sources of NL Surface Forms

Freebase links entities to NL Surface forms:

- Wikipedia
- MusicBrainz
- IMDB
- And many more…



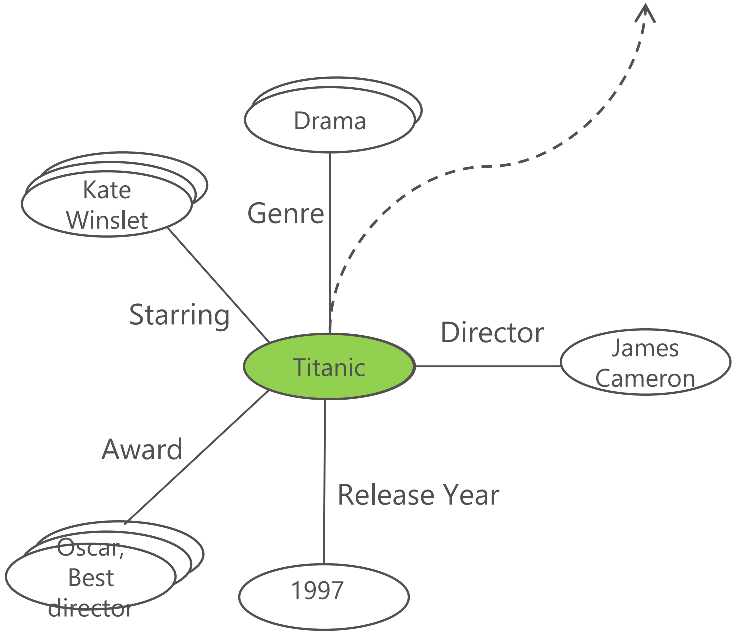
Article Talk Read View source View hi

Titanic (1997 film)

From Wikipedia, the free encyclopedia

Titanic is a 1997 American epic romantic disaster film directed, written, co-produced, and co-edited by James Cameron. A fictionalized account of the sinking of the RMS *Titanic*, it stars Leonardo DiCaprio and Kate Winslet as members of different social classes who fall in love aboard the ship during its ill-fated maiden voyage.

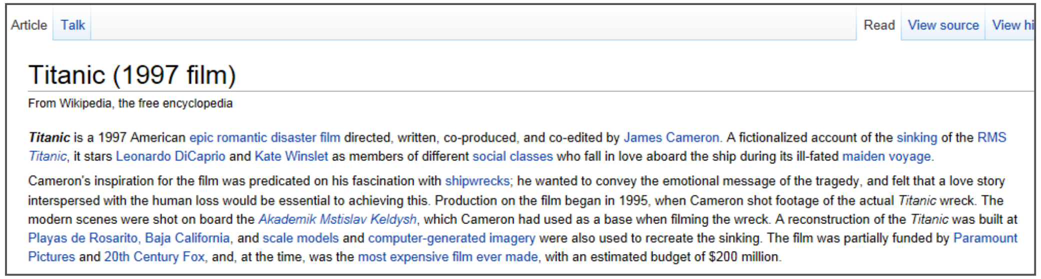
Cameron's inspiration for the film was predicated on his fascination with shipwrecks; he wanted to convey the emotional message of the tragedy, and felt that a love story interspersed with the human loss would be essential to achieving this. Production on the film began in 1995, when Cameron shot footage of the actual *Titanic* wreck. The modern scenes were shot on board the *Akademik Mstislav Keldysh*, which Cameron had used as a base when filming the wreck. A reconstruction of the *Titanic* was built at Playas de Rosarito, Baja California, and scale models and computer-generated imagery were also used to recreate the sinking. The film was partially funded by Paramount Pictures and 20th Century Fox, and, at the time, was the most expensive film ever made, with an estimated budget of $200 million.



Unsupervised Data Mining with KGs

Step #3: Annotate with 1st Order Relations

Titanic	B-film_name
stars	O
Leonardo	B-film_starring
Dicaprio	I-film_starring
and	O
Kate	B-film_starring
Winslet	I-film_starring
as	O
…	…

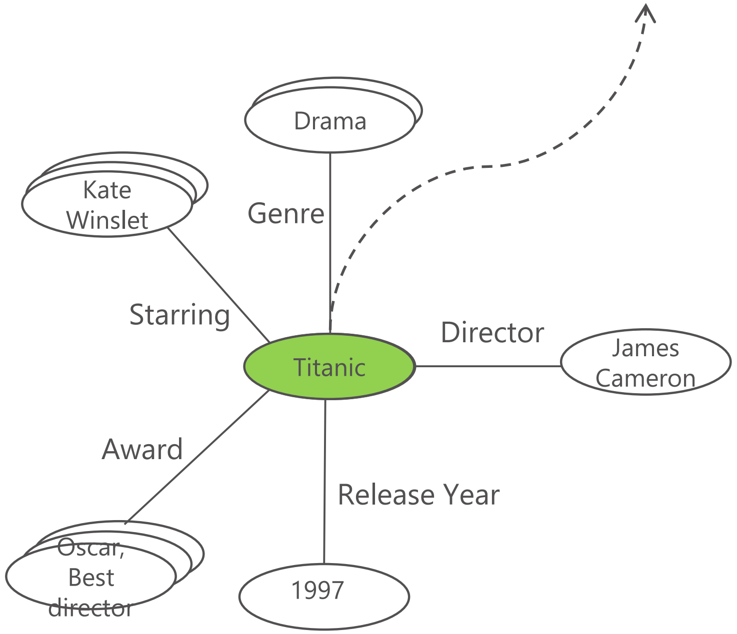

Article Talk Read View source View hi

Titanic (1997 film)

From Wikipedia, the free encyclopedia

Titanic is a 1997 American epic romantic disaster film directed, written, co-produced, and co-edited by James Cameron. A fictionalized account of the sinking of the RMS *Titanic*, it stars Leonardo DiCaprio and Kate Winslet as members of different social classes who fall in love aboard the ship during its ill-fated maiden voyage.

Cameron's inspiration for the film was predicated on his fascination with shipwrecks; he wanted to convey the emotional message of the tragedy, and felt that a love story interspersed with the human loss would be essential to achieving this. Production on the film began in 1995, when Cameron shot footage of the actual *Titanic* wreck. The modern scenes were shot on board the *Akademik Mstislav Keldysh*, which Cameron had used as a base when filming the wreck. A reconstruction of the *Titanic* was built at Playas de Rosarito, Baja California, and scale models and computer-generated imagery were also used to recreate the sinking. The film was partially funded by Paramount Pictures and 20th Century Fox, and, at the time, was the most expensive film ever made, with an estimated budget of $200 million.



Unsupervised Data Mining with KGs

Step #4: Instantiate All Entities of CPN Type

Explore “depth” of entity-type

→ large entity lists (gazetteers)

Article | Talk | Read | Edit source | View hi

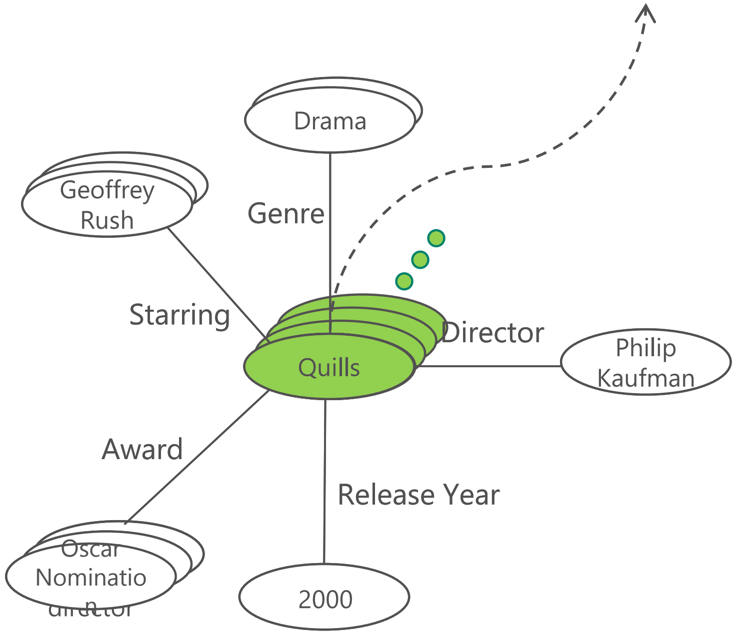
Quills

From Wikipedia, the free encyclopedia

This article is about the film. For the writing utensils, see Quill. For other uses, see Quill (disambiguation).

Quills is a 2000 period film directed by Philip Kaufman and adapted from the Obie award-winning play by Doug Wright, who also wrote the original screenplay. Inspired by the life and work of the Marquis de Sade, *Quills* re-imagines the last years of the Marquis' incarceration in the insane asylum at Charenton. It stars Geoffrey Rush as the Marquis de Sade, Joaquin Phoenix as the Abbé du Coulmier, Michael Caine as Dr. Royer-Collard, and Kate Winslet as laundress Madeleine "Maddie" LeClerc.

Well received by critics, *Quills* garnered numerous accolades for Rush, including nominations for an Oscar and a Golden Globe. The film was a modest art house success, averaging $27,709 per screen its debut weekend, and eventually grossing $17,989,277 internationally. Cited by historians as factually inaccurate, *Quills* filmmakers and writers said they were



Unsupervised Data Mining with KGs

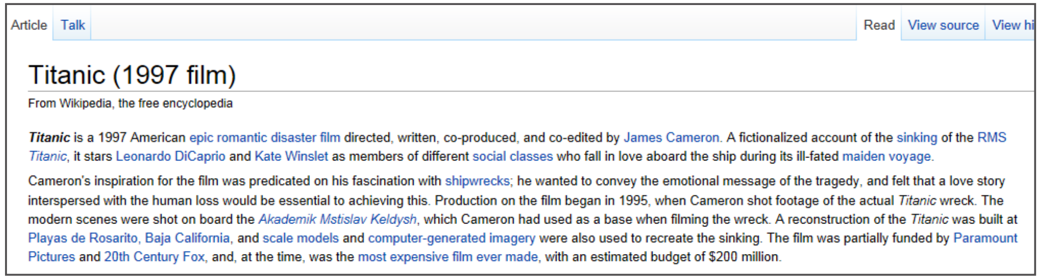
Step #5: Get 2nd Order Relations

Knowledge graph "compositionality"

- Entity-relation templates (grammars) can be composed

Template	Frequency
ent	44.9%
$type \sqcap rel(ent)$	12.8%
$ent_0 \sqcap rel(ent_1)$	7.7%
$ent \sqcap type$	5.8%
$type$	5.8%
$attr(ent)$	3.8%
$ent_1 \sqcap rel(ent_0)$	3.2%
$rel(ent)$	1.9%
$ent_0 \sqcap rel(ent_1, rel(ent_2))$	1.3%
$type_1 \sqcap rel(type_0)$	1.3%

Ten most frequently occurring templates among entity-based queries (Pound et al., CIKM'12)



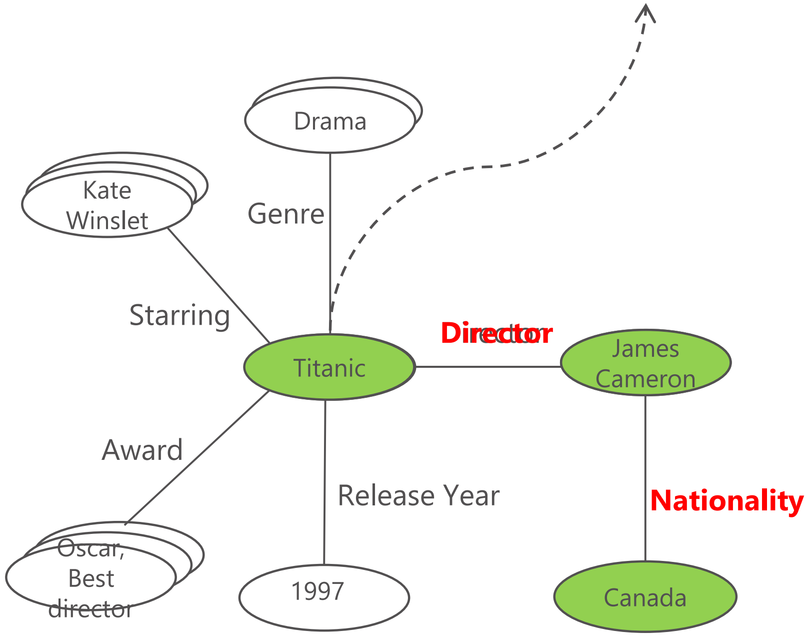
Article Talk Read View source View hi

Titanic (1997 film)

From Wikipedia, the free encyclopedia

Titanic is a 1997 American epic romantic disaster film directed, written, co-produced, and co-edited by James Cameron. A fictionalized account of the sinking of the RMS *Titanic*, it stars Leonardo DiCaprio and Kate Winslet as members of different social classes who fall in love aboard the ship during its ill-fated maiden voyage.

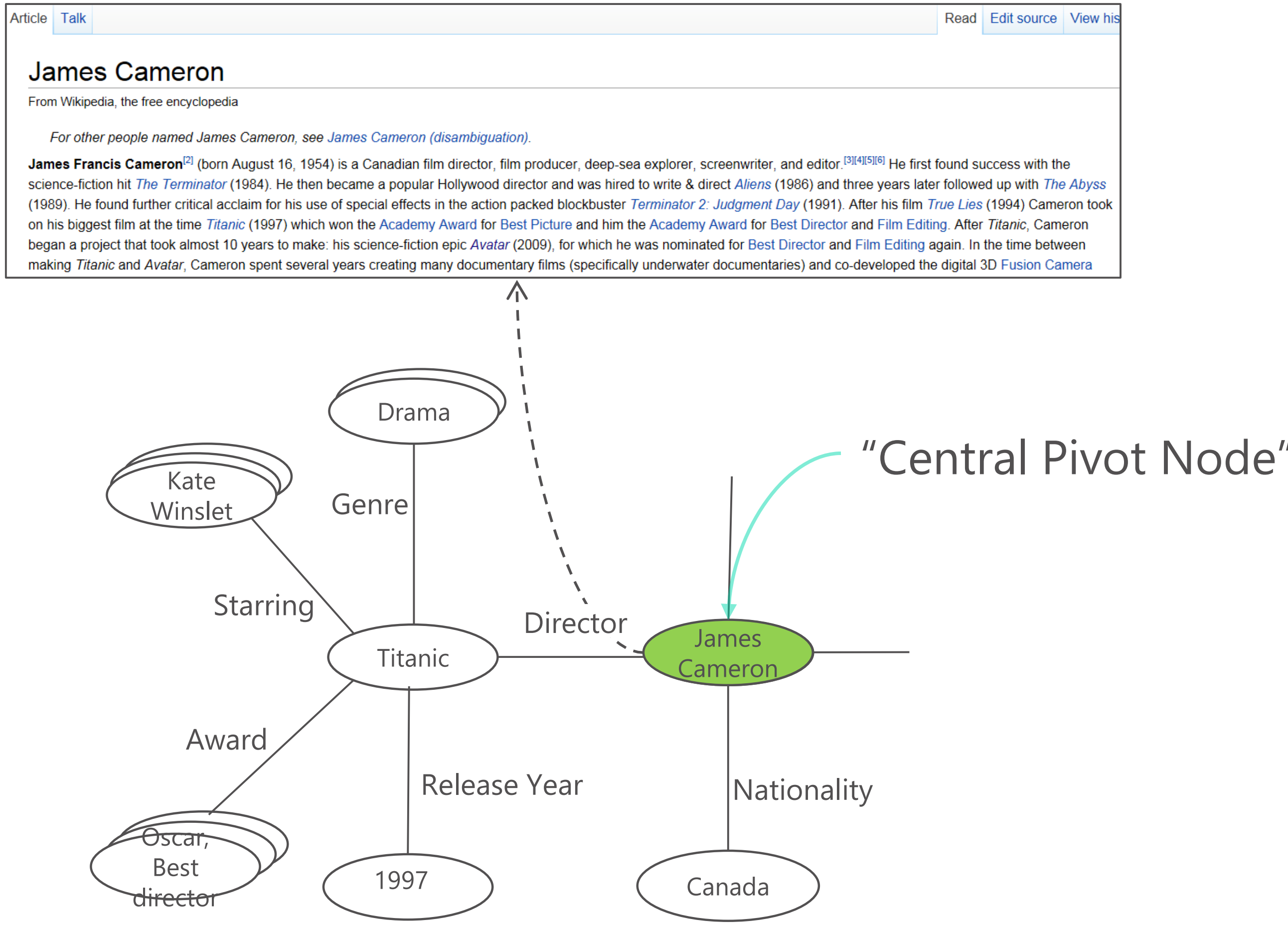
Cameron's inspiration for the film was predicated on his fascination with shipwrecks; he wanted to convey the emotional message of the tragedy, and felt that a love story interspersed with the human loss would be essential to achieving this. Production on the film began in 1995, when Cameron shot footage of the actual *Titanic* wreck. The modern scenes were shot on board the *Akademik Mstislav Keldysh*, which Cameron had used as a base when filming the wreck. A reconstruction of the *Titanic* was built at Playas de Rosarito, Baja California, and scale models and computer-generated imagery were also used to recreate the sinking. The film was partially funded by Paramount Pictures and 20th Century Fox, and, at the time, was the most expensive film ever made, with an estimated budget of $200 million.



Unsupervised Data Mining with KGs

Step #6: Select New CPN and Repeat (Crawl Graph)

- Select new central pivot node
- Repeat steps #1-5
- Crawl graph until complete



Experimental Setup

- Scenario: *developer seeks to build a NL movie search application (Netflix)*
- Training
  - Freebase film (movies) domain, 56 relations with linked Wikipedia articles
  - 4 Netflix properties: movies (175K), actors (234K), genres (685), directors (59K)
  - 10K NL surface forms = Wikipedia ("Meg Ryan starred with Tom Hanks in …")
- Testing: 2 Conditions
  - Mined Testset
    - Development corpus
    - 1K Wikipedia sentences
    - "Matched" condition
  - Control Testset
    - Target Netflix (true) testset
    - 2K utterances from user data collection

Results

	Manual Transcriptions					ASR Output				
	Movie	Actor	Genre	Director	All	Movie	Actor	Genre	Director	All
Supervised										
CRF Lexical + Gazetteers	51.25%	86.29%	93.26%	64.86%	66.53%	45.15%	82.56%	88.58%	58.59%	60.96%
CRF Lexical only	46.44%	80.22%	92.83%	52.94%	61.72%	39.21%	74.86%	86.21%	45.36%	54.10%
Unsupervised										
Gazetteers only	69.69%	50.70%	15.76%	2.63%	51.14%	59.66%	47.78%	11.80%	2.82%	43.88%
CRF Lexical only	0.19%	9.67%	0.00%	62.83%	5.61%	0.20%	9.67%	0.00%	57.14%	5.27%
+ Gazetteers	1.96%	72.35%	4.73%	79.03%	31.94%	1.74%	69.76%	3.57%	75.00%	30.77%

Mismatched Style of training (Wikipedia) and testing (Netflix) significantly impacting results

Leveraging KGs for Semantic Parsing

Procedure

- Unsupervised Data Mining with Knowledge Graphs
  - 6 step procedure
  - Auto-annotated (unsupervised) data used to train SLU
- Style Adaptation
- Modeling Relations for Semantic Parsing

Adaptation

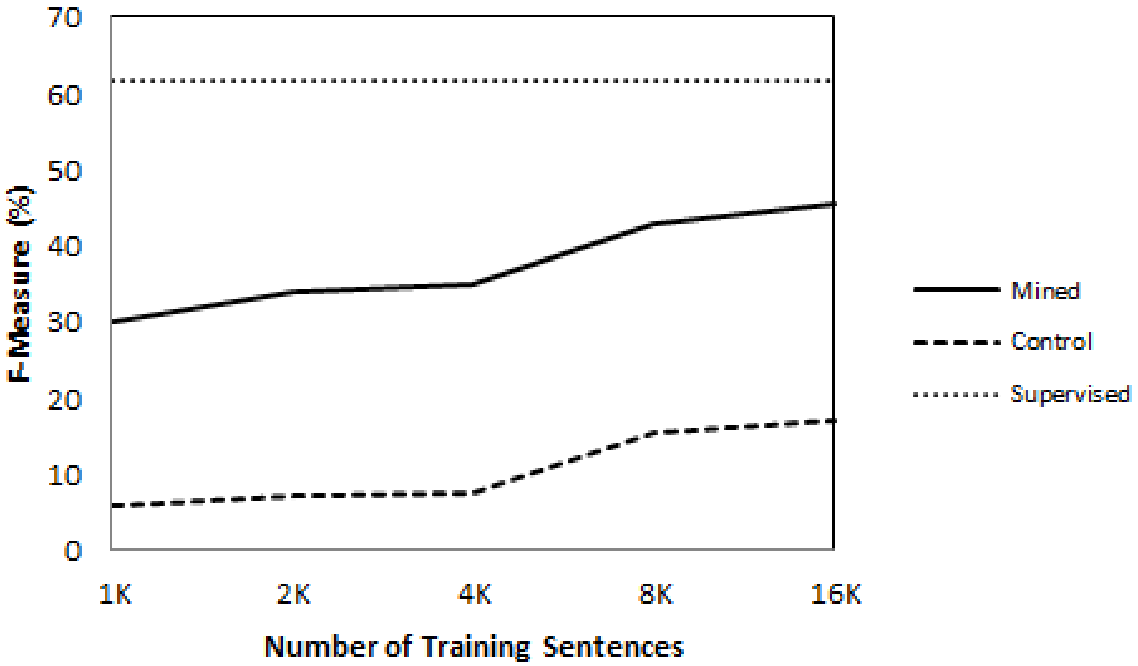
Addressing Mismatch Problem

- Mismatch between training/testing can occur if:
  - **Genre Differences**: NL surface forms from knowledge graph sources *mismatch* with the target genre
  - **Poor Coverage**: sparse set of surface forms for tail patterns
- Results: Freebase + Wikipedia
  - **Training:** single source of NL surface forms = Wikipedia ("Meg Ryan starred with Tom Hanks in …")
  - **Testing:** Netflix movie search ("show me some funny flicks with Meg Ryan")

Adaptation (cont.)

Addressing Mismatch Problem

- Solution
  - Rely on relative robustness to mismatch of Gazetteers
  - Unsupervised MAP-like bootstrap/retraining adaptation
  - Adapt to representative sample of data from target domain



Mined = Wikipedia-Wikipedia
Control = Wikipedia-Netflix

Results

	Manual Transcriptions					ASR Output				
	Movie	Actor	Genre	Director	All	Movie	Actor	Genre	Director	All
Supervised										
CRF Lexical + Gazetteers	51.25%	86.29%	93.26%	64.86%	66.53%	45.15%	82.56%	88.58%	58.59%	60.96%
CRF Lexical only	46.44%	80.22%	92.83%	52.94%	61.72%	39.21%	74.86%	86.21%	45.36%	54.10%
Unsupervised										
Gazetteers only	69.69%	50.70%	15.76%	2.63%	51.14%	59.66%	47.78%	11.80%	2.82%	43.88%
CRF Lexical only	0.19%	9.67%	0.00%	62.83%	5.61%	0.20%	9.67%	0.00%	57.14%	5.27%
+ Gazetteers	1.96%	72.35%	4.73%	79.03%	31.94%	1.74%	69.76%	3.57%	75.00%	30.77%
+ Adaptation	71.72%	58.61%	29.55%	77.42%	60.38%	55.74%	62.70%	30.95%	73.21%	54.69%

Leveraging KGs for Semantic Parsing

Procedure

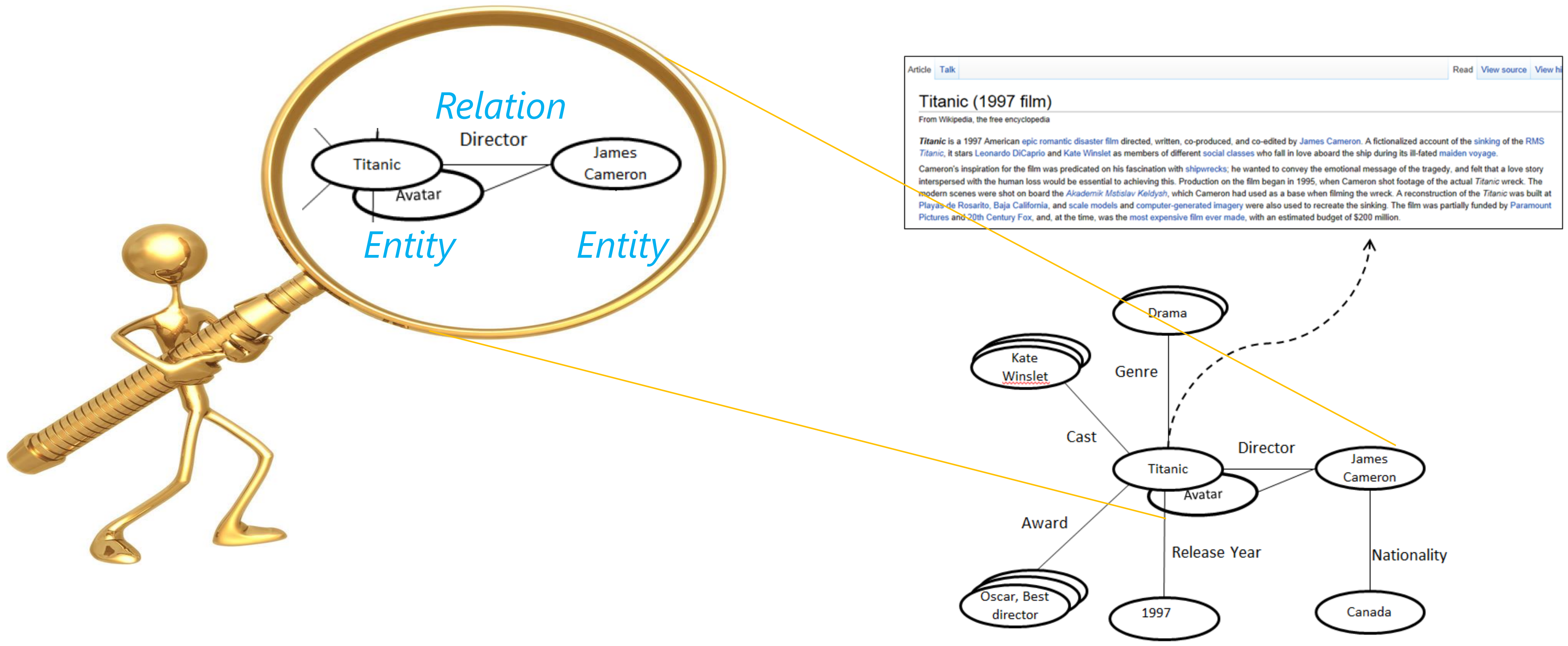
- Unsupervised Data Mining with Knowledge Graphs
  - 6 step procedure
  - Auto-annotated (unsupervised) data used to train SLU

- Style Adaptation

- Modeling Relations for Semantic Parsing

Modeling *Relations* for Semantic Parsing

Semantic Templates



Modeling Relations for Semantic Parsing

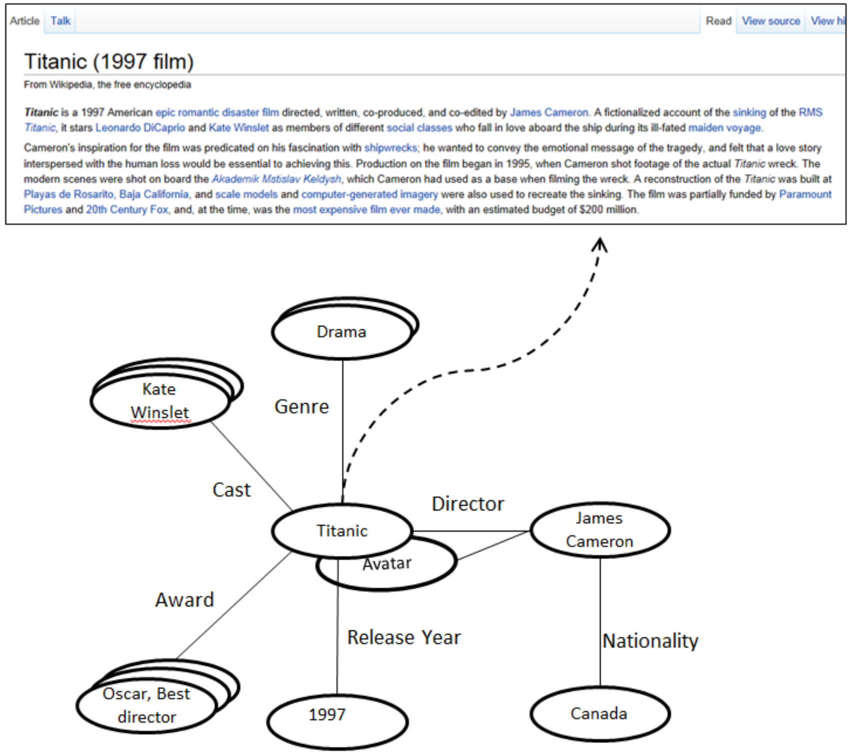
Semantic Templates

Extracted entities provide foundation for higher-level (grammatical) structure

- Leverage our prior work* to identify *entity-relation patterns*
- Induce grammars from templates
- "Repair" missing entities (e.g., "show me *movies with* ______")

Template	Frequency
ent	44.9%
$type \sqcap rel(ent)$	12.8%
$ent_0 \sqcap rel(ent_1)$	7.7%
$ent \sqcap type$	5.8%
$type$	5.8%
$attr(ent)$	3.8%
$ent_1 \sqcap rel(ent_0)$	3.2%
$rel(ent)$	1.9%
$ent_0 \sqcap rel(ent_1, rel(ent_2))$	1.3%
$type_1 \sqcap rel(type_0)$	1.3%

Ten most frequently occurring templates among entity-based queries (Pound et al., CIKM'12)



* Dilek Hakkani-Tur, Larry Heck, and Gokhan Tur, Using a Knowledge Graph and Query Click Logs for Unsupervised Learning of Relation Detection, ICASSP 2013

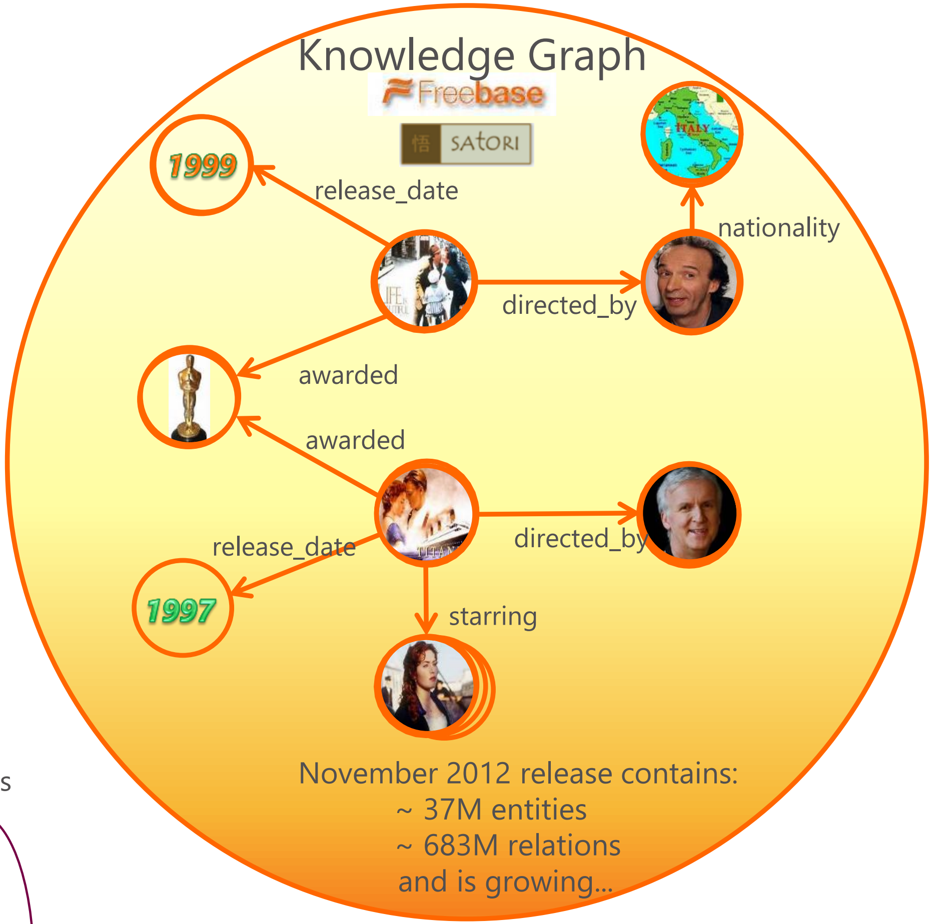
Results

Summary

	Manual Transcriptions					ASR Output				
	Movie	Actor	Genre	Director	All	Movie	Actor	Genre	Director	All
Supervised										
CRF Lexical + Gazetteers	51.25%	86.29%	93.26%	64.86%	66.53%	45.15%	82.56%	88.58%	58.59%	60.96%
CRF Lexical only	46.44%	80.22%	92.83%	52.94%	61.72%	39.21%	74.86%	86.21%	45.36%	54.10%
Unsupervised										
Gazetteers only	69.69%	50.70%	15.76%	2.63%	51.14%	59.66%	47.78%	11.80%	2.82%	43.88%
CRF Lexical only	0.19%	9.67%	0.00%	62.83%	5.61%	0.20%	9.67%	0.00%	57.14%	5.27%
+ Gazetteers	1.96%	72.35%	4.73%	79.03%	31.94%	1.74%	69.76%	3.57%	75.00%	30.77%
+ Adaptation	71.72%	58.61%	29.55%	77.42%	60.38%	55.74%	62.70%	30.95%	73.21%	54.69%
+ Relations				84.62%	61.02%				80.67%	55.40%

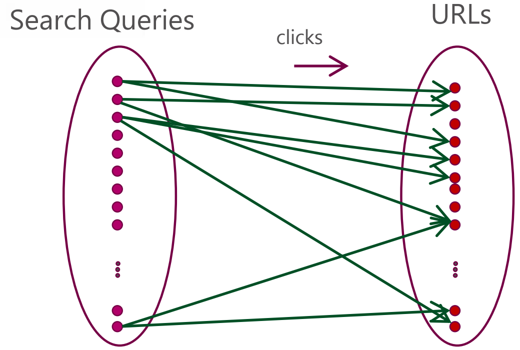
Linking Data to the Knowledge Graph

- Find queries whose users clicked on URLs found in steps 1 and 2
- Transfer labels from the graph to these queries



3

Query Click Logs

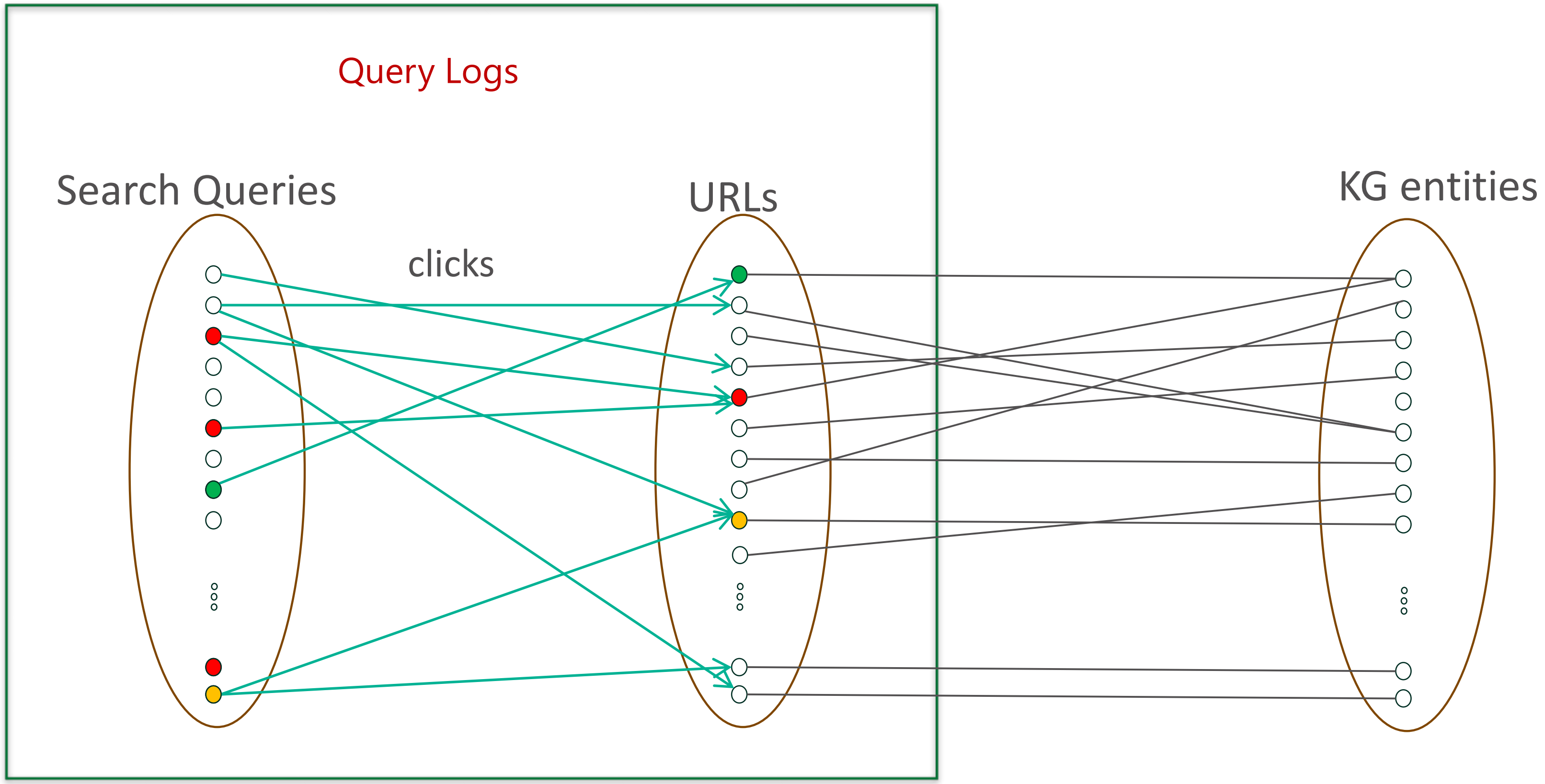


Who directed the movie Life is beautiful
Director of Life is beautiful
...

More details in:

- Gokhan Tur, Minwoo Jeong, Ye-Yi Wang, Dilek Hakkani-Tur, and Larry Heck, Exploiting the Semantic Web for Unsupervised Natural Language Semantic Parsing, in *Proc. Interspeech*, September 2012.
- Dilek Hakkani-Tur, Larry Heck, and Gokhan Tur, Using a Knowledge Graph and Query Click Logs for Unsupervised Learning of Relation Detection, IEEE ICASSP, May 2013.

Search Logs and Knowledge Graph: Tripartite Graph



Combining Knowledge Graph with Search Query Logs

Semantic Parsing of Structured Web Pages

structured document

Id	Name	City	State
r_1_3671	wild ginger	seattle	washington

\+

query log

wild ginger restaurant in seattle wa

AutoLabel

automatically annotated query

<restaurant parse="wild ginger restaurant in seattle wa">
<name start="0" end="2">wild ginger</name>
<city start="4" end="5">seattle</city>
<state start="5" end="6">wa</state>
</restaurant>

Combining Knowledge Graph with Search Query Logs

structured document

Id	Movie Name	Director	Genre
1392170	the hunger games	gary ross	action; drama; sci-fi; thriller

Semantic Parsing of Structured Web Pages

+

query log

full cast of hunger games with jennifer lawrence

AutoLabel

automatically annotated query

<movies parse="full cast of hunger games with jennifer lawrence">
<name>hunger games</name>
<actor>jennifer lawrence</actor>
</movies>

Slot Filling Experiments
[Tur et al., Interspeech 2012]

- Focused on bootstrapping SLU model for the movies domain with 4 named slots (movie, actor, director, and character names)
- Used only the queries where there is a non-slot stopword (*remove non-NL*)
  - discard "~~*avatar*~~" or "~~*avatar cameron*~~" keep "*avatar by cameron*"
  - discard "~~*the artist movie*~~" keep "*reviews for the movie artist*"
- Did not use queries where there is a partial untagged match (*remove noise*)
  - discard "~~*the artist by michel hazanavic*~~" keep "*the artist by michel hazanavicius*"
- Control supervised set:
  - Train: 2,700 manually annotated sentences
  - Test: 300 manually annotated sentences
  - 3,750 slots (about 1,400 movie names).

	Movie Name	Actor Name	All Slots
Supervised	55.22%	81.25%	64.26%
All	38.39%	89.13%	48.94%
NL-Like	47.94%	84.26%	**57.73%**

What if there is some in-domain data?

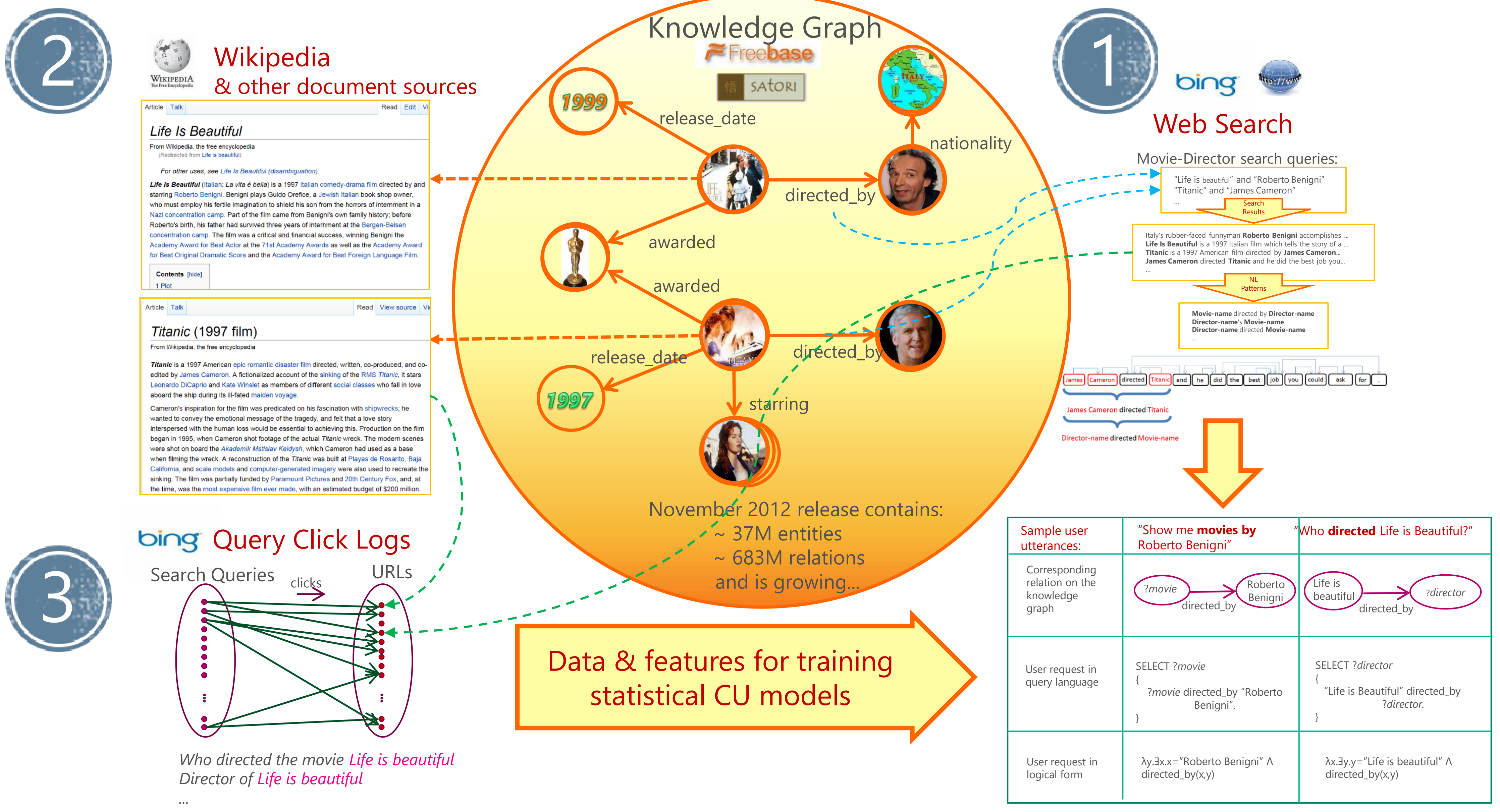
- … but not annotated.
- Idea: Use a variant of unsupervised MAP adaptation.
- Annotate the data, *U*, with the bootstrap model, *M*.

$$T_U = argmax_T\, P_M(T|U)$$

- Weighted interpolation of models, or
  - Simple data concatenation
  - Only using the in-domain data with automated annotations

	Movie Name	Actor Name	All Slots
Supervised	55.22%	81.25%	64.26%
All	38.39%	89.13%	48.94%
NL-Like	47.94%	84.26%	57.73%
NL-Like + Unlabeled Set	50.21%	85.47%	**60.03%**

Scaling Conversational Understanding Systems



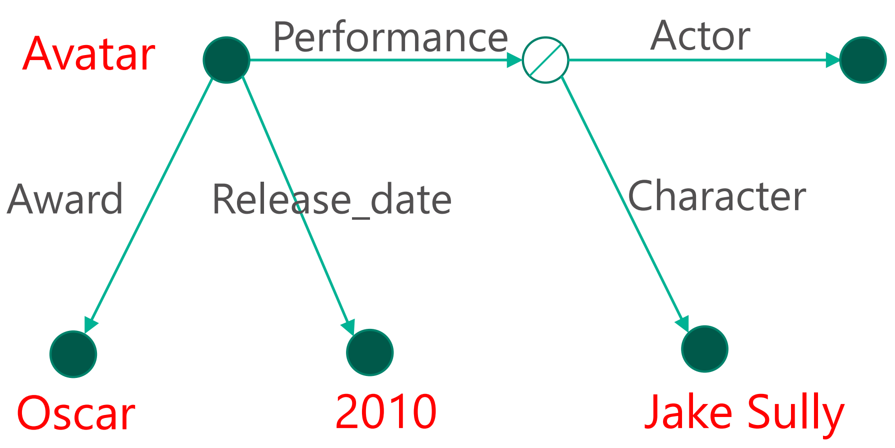
Outline

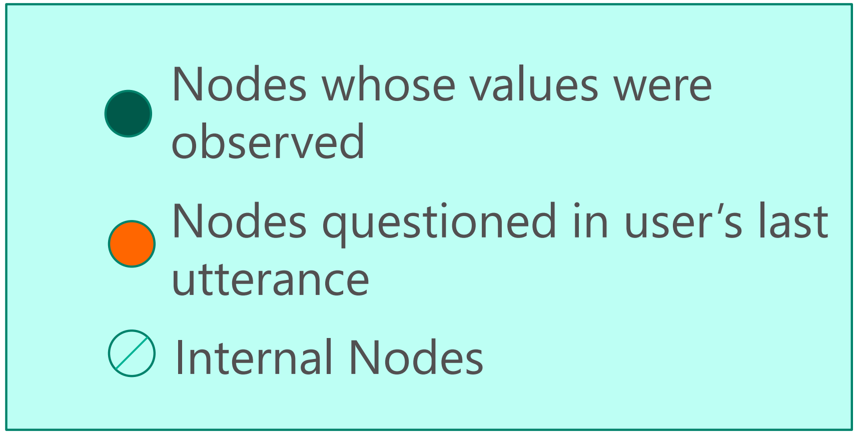
- Conversational systems
- Sematic Web and Linked Data Sources
- SLU based on knowledge graphs
- Leveraging Context with knowledge graphs
- **Learning from the Semantic Web**
  - Learning Entity Extraction and Relation Detection from Data Linked to Knowledge Graphs
    Examples for learning the mappings from utterances to relations and entities
  - New Intent Discovery
    Relations or transactions missing on the graph
  - Introducing Weights to the Knowledge Graph (special case: entity types)
  - Deep Learning from Knowledge Graphs

Back to Example 1

U1: Who played Jake Sully in the Oscar winning 2010 movie Avatar

U2: Play this movie.

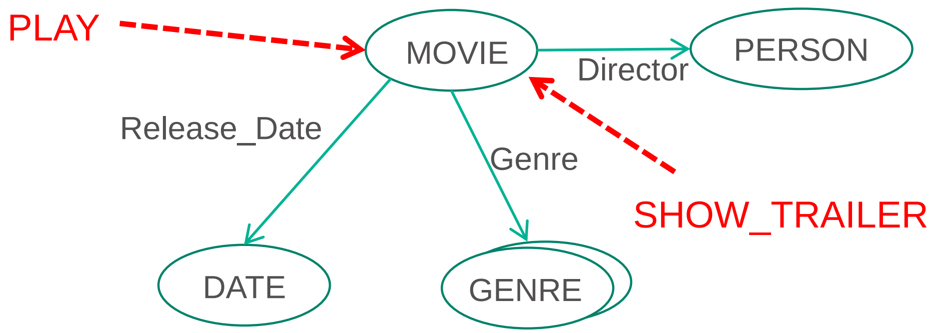




New Intent Discovery

- Information captured in the KG may not always be sufficient for conversational understanding, especially for **transactional** intents.

- Automatically capture and tag semantic information that includes **new** user intents and related concepts in real human-machine dialogs.



- Search query click logs [LinEtAl, 2012]

Related Work: Sequence Clustering & Labeling
[Cheung and Li; WSDM,2012]

- An unsupervised method for:
  - clustering (web search) queries with similar intent and
  - producing a pattern (sequence of semantic concepts and/or lexical items) for each intent.
- Steps:

1. **Feature extraction:** A query is represented by N*M matrix.

   N: number of features per word (N=S+L)

   S (L): number of semantic (lexical) features

   M: the number of n-grams in the query

$$\begin{array}{rc} & \begin{array}{cccc} \mathit{2010} & \mathit{buick} & \mathit{regal} & \mathit{review} \end{array} \\ \begin{array}{r} \text{[car]} \\ \text{[episode]} \\ \text{[model]} \\ \text{[season]} \\ \text{[year]} \\ \ldots \\ \text{review} \\ \text{test} \end{array} & \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

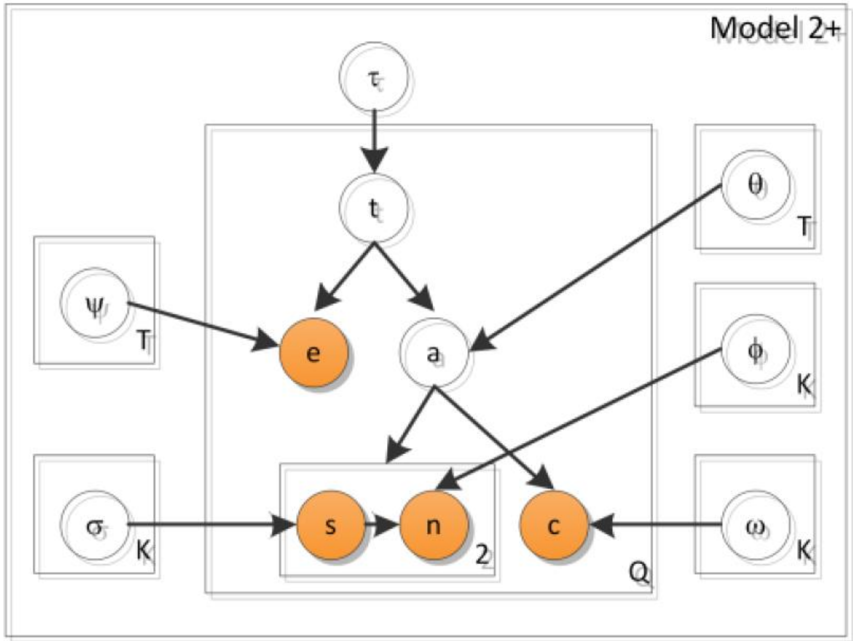
Semantic features are entity types from Freebase, the corresponding value is 1 if the n-gram is marked as an entity of specific type in Freebase.

Related Work: Sequence Clustering & Labeling
[Cheung and Li; WSDM,2012]

- Steps (cont):
  2. **Sequence Clustering:** Agglomerative clustering with using a distance metric based on dynamic time warping (DTW) between a pair of sequences.
  3. **Intent Summarization:** produces a pattern that describes the intent of each query cluster.
  4. **Unseen instance annotation:** Patterns are also represented as N*M matrices, allowing for computation of most similar patterns for a new query.
- 10 domains, 125K queries, discovered 1400 intents.
  - 90% of the discovered intents and 80% of annotated queries were judged correct by majority of (crowd) annotators.
  - Domain coverage: above 20% for certain frequent domains.

Related Work: Actions for Entity Centric Search
[Lin Et Al 2012]

- Majority of search queries contain an entity.
- Entity Centric Search: pair entities with actions that can be performed on them.
- Finding such actions is viewed as the problem of probabilistic inference in a graphical model that captures how an entity bearing query is generated.
- Specialized for web search and what is already known is not used.



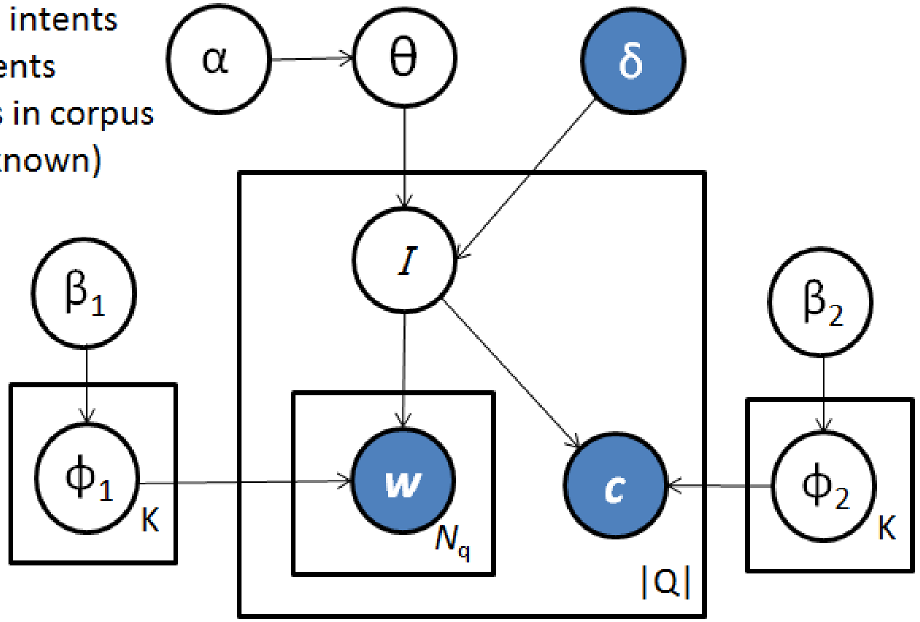
Actions recommended for the query: Webster University

1. Find address
2. See pictures of
3. Find map of
4. Read news about
5. Apply for jobs at
6. See cost of
7. See ranking of

New Intent Discovery

- Motivation: Users type in a query and click on a link, both of which are related to their intent.
- A novel Bayesian hierarchical model that **can exploit what is already known** during training.

I_g: fixed - known (given) intents
I_u: newly discovered intents
$|Q|$: number of queries in corpus
$I = I_g \cup I_u$ (g:given ; u:unknown)
$K = |I_g|+|I_U|$



[HakkaniTurEtAl, 2013]

Inference and Learning

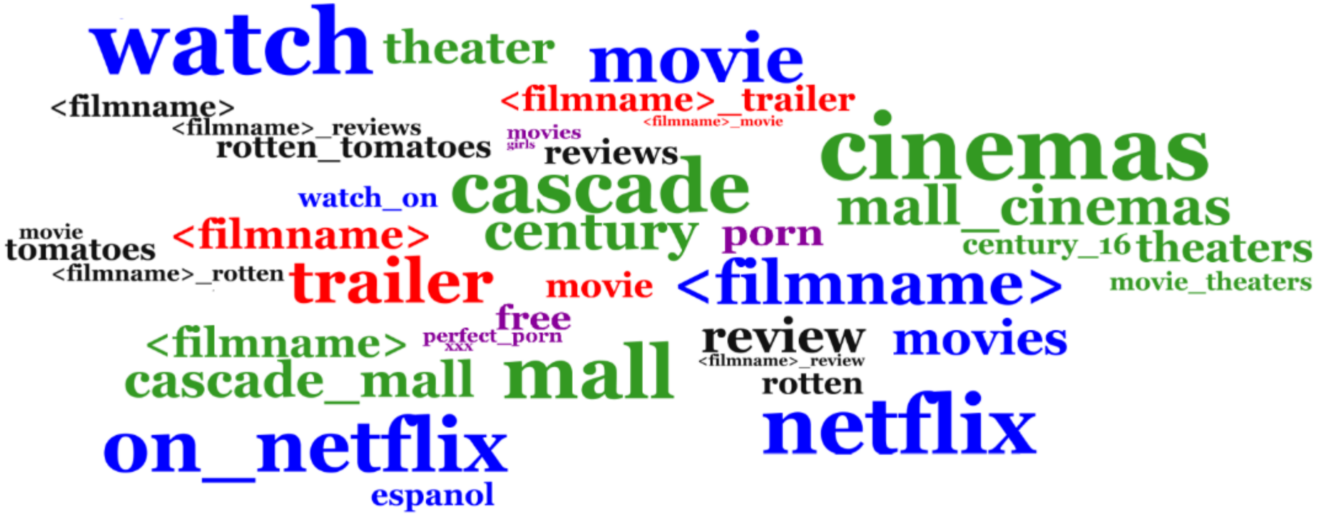
- Gibbs sampling
- For each query q, if no prior intent information is available, we sample an intent given its click, words and the hyper-parameters:

$$p(I_q = k|\mathbf{c}, \mathbf{w}_q, I_{-q}, \alpha, \beta_1, \beta_2, \alpha) \propto$$
$$\frac{n_q^k + \alpha}{(|Q| - 1 + K\alpha)} * \frac{n_c^k + \beta_2}{n_{(.)}^k + |C|\beta_2} * \prod_{i=1}^{N_q} \frac{n_{w_i}^k + \beta_1}{n_{(.)}^k + |V|\beta_1}$$

- n_q^k is the number of queries assigned to a semantic class k excluding the query q
- n_c^k is the number of times c is assigned to intent class k
- $n_{w_i}^k$ is the number of times word w_i is assigned to class k
- (.) indicates sum over the object, i.e., words.

New Intent Discovery

- Outcome: latent intent clusters
- Used to enrich relation detection models to integrate new intents.



Movie reviews
Movie theater
Play movie
Movie trailer
…

More details in:
Dilek Hakkani-Tur, Asli Celikyilmaz, Larry Heck, and Gokhan Tur, A Weakly-Supervised Approach for Discovering New User Intents from Search Query Logs, *Proc. Interspeech*, August 2013

Relation Detection with New Intents

- **Query Click Logs (QCL) data:** 81K queries, 8K movie related URLs.
- **Development and Test Sets:**
  - 6000 and 5706 user turns from movies (Netflix data sets).
  - On average, 0.98 relations per utterance (example utterance that does not invoke any KG relation: "*gone with the wind*").
- **Bootstrap model:** 7 relations from the KG, cover 47% of the annotated relations in the development and test sets.
- 7 clusters discovered by our new intent discovery model and validated.
  - 4 (*movie review, movie content, play movie, play trailer of movie*) were similar to categories covered in the test sets.
  - 3 were not included in the development and test sets (i.e., *find theater*).

Experiments

Model	Dev		Test	
	MF-known	MF-new	MF-known	MF-new
Bootstrap	52.1%	11.7%	48.0%	12.0%
CIM	52.6%	36.7%	**49.1%**	**36.9%**
Crowd-Sup.	42.7%	39.4%	45.5%	42.1%

- **MF-{known, new}:** macro-averaged per class F-measure for previously known and newly discovered intent categories.
- ***CIM*** includes labels from the bootstrap models as automatic supervision.
- In ***CIM***, word sequences corresponding to entities in a gazetteer were canonicalized by replacing them with the entity type.

Ongoing/Future Work

- Adding latent features to supervised models [JiEtAl, 2014]
  - Benefiting from large amounts of queries in query click logs

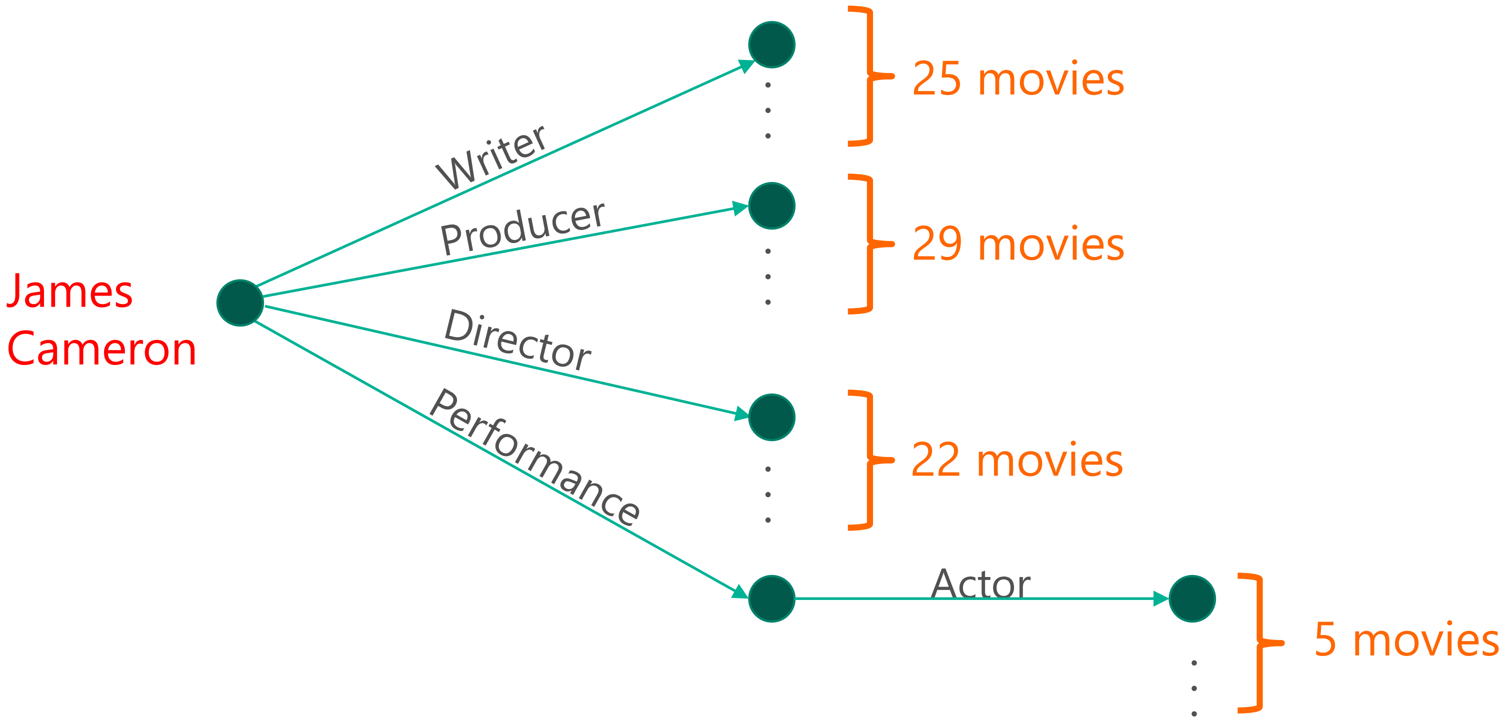
- New graphical modeling approaches
  - i.e. Relational models, Riedel et al. 2012

Outline

- Conversational systems
- Sematic Web and Linked Data Sources
- SLU based on knowledge graphs
- Leveraging Context with knowledge graphs
- **Learning from the Semantic Web**
  - Learning Entity Extraction and Relation Detection from Data Linked to Knowledge Graphs
    Examples for learning the mappings from utterances to relations and entities
  - New Intent Discovery
    Relations or transactions missing on the graph
  - Introducing Weights to the Knowledge Graph (special case: entity types)
  - Deep Learning from Knowledge Graphs

Back to Example 2

U1: Find some action movies by James Cameron



- Knowledge in the graph is not probabilistic.

Introducing weights to the KG: entity types

- Lists of entities (such as gazetteers of movie and actor names) commonly used for interpretation of natural language user queries in dialog systems [Raymond&Riccardi, 2007; HillardEtAl, 2011; TurEtAl,2011].

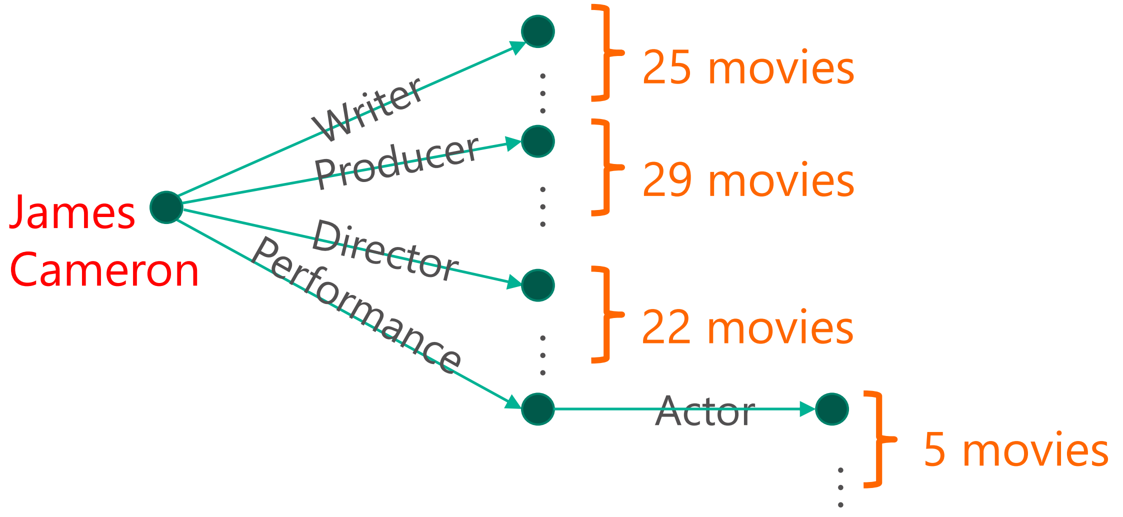
- [HillardEtAl, 2011] based on search queries and clicked URLs to assign a weight to each entity term to estimate if a term is more commonly used in natural language as an entity or not.
  - Good for types represented on the web and search queries.
  - Doesn’t solve the problem in the previous example.

Counts from the Knowledge Graph

- Use relative frequencies from the populated knowledge graph:

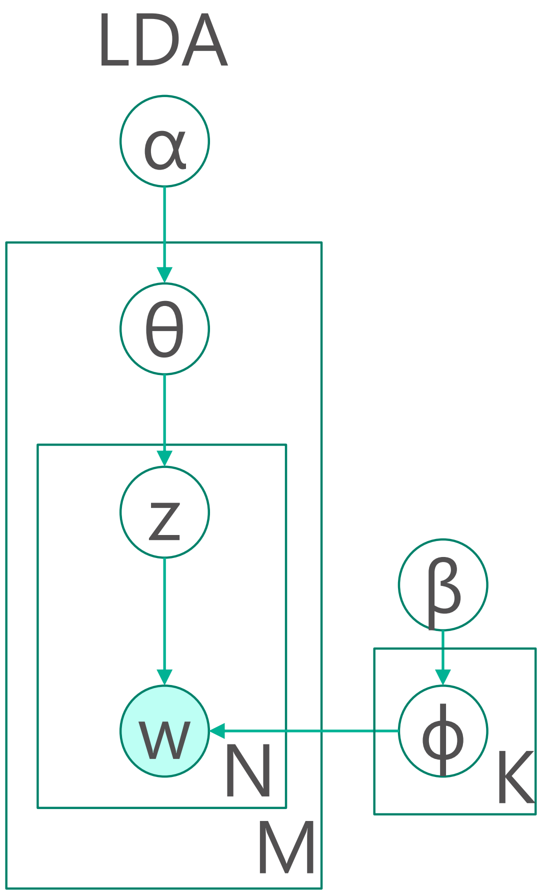
$$P_{SG}(t_j|e_i) = \frac{C(e_i, t_j)}{\sum_{k \in T} C(e_i, t_k)}$$

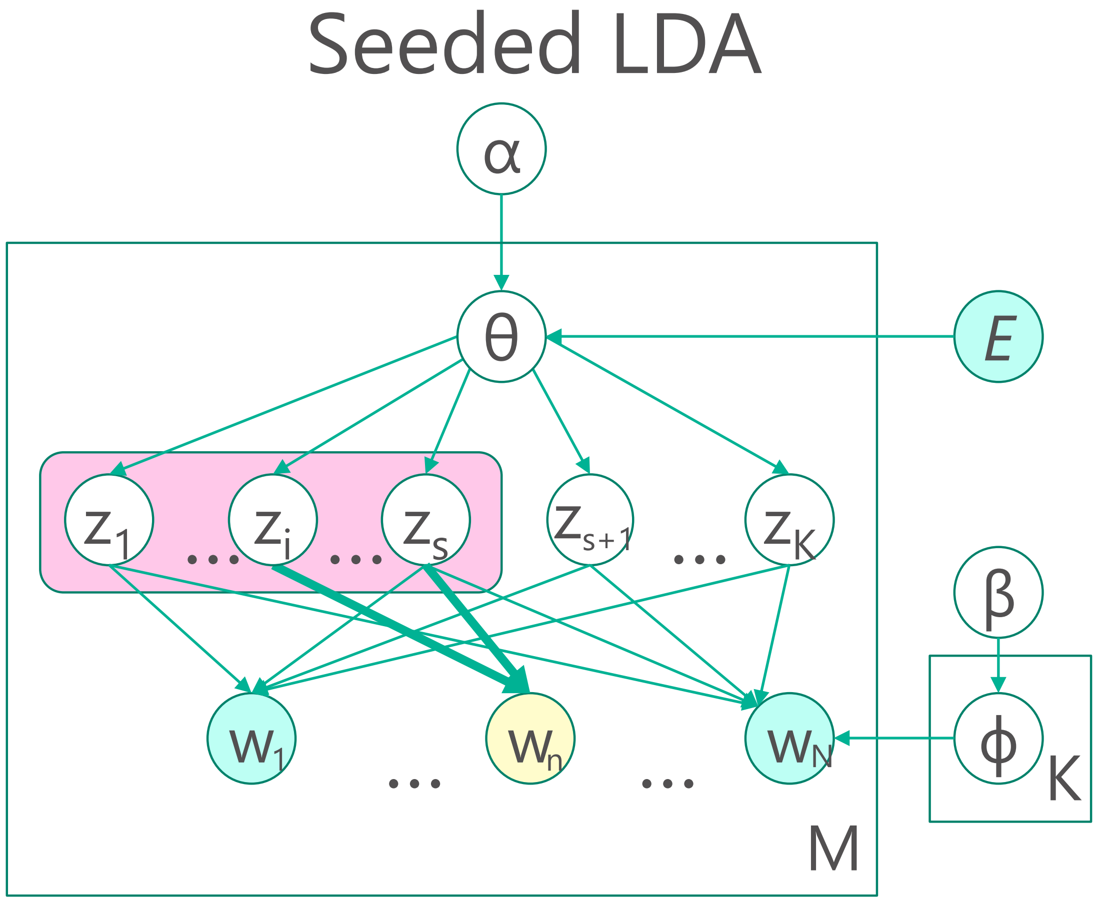
- e_i: entity i
- t_j: type j
- Relies on complete knowledge on the graph.



P(writer | James Cameron) = 25 / 81

Seeded Latent Dirichlet Allocation





[HakkaniTurEtAl, 2014]

Natural Language Query to SPARQL Query Data Sets

Query Statistics	Training	Test
No. with SPARQL annotations	3,338	1,086
% with no relation (i.e. entity only)	10.1%	9.1%
% with 1 relation	70.4%	69.2%
% with 2 relations	10.2%	10.7%
% with 3 or more relations	1%	1.6%
% not covered by Freebase	8.3%	9.4%

- Statistics different than the one for web search queries, where close to half of the queries include a single entity [PoundEtAl, 2012].

- Data: http://research.microsoft.com/en-us/projects/kgandld4cu/

Experiments

Number of examples in the training set

- F-measure averaged over 9 training and test splits.
- KG-weighted requires a fully populated knowledge graph.
- SLDA trained on Wikipedia movie documents.

Outline

- Conversational systems
- Sematic Web and Linked Data Sources
- SLU based on knowledge graphs
- Leveraging Context with knowledge graphs
- **Learning from the Semantic Web**
  - Learning Entity Extraction and Relation Detection from Data Linked to Knowledge Graphs
    Examples for learning the mappings from utterances to relations and entities
  - New Intent Discovery
    Relations or transactions missing on the graph
  - Introducing Weights to the Knowledge Graph (special case: entity types)
  - Deep Learning from Knowledge Graphs

Deep Learning from Knowledge Graphs

Transforming knowledge into deep neural representations

KGs *enriched* through data mining

- search queries-clicks, captions (snippets)
- Wikipedia pages

➔ *massive, structured, labeled data*

Deep learning can discover compact semantic space of knowledge from enriched KGs

Knowledge Embeddings

Knowledge Embeddings provide computable vector-space of semantic relations in R^N

$$P(e_i\ R_{ij}\ e_j)$$

Deep Learning from Knowledge Graphs

Newly Emerging Approaches in the Literature

Bordes, Chopra, & Weston (2014)

- Embedding model for entity-bearing queries in open QA
- **Supervised** training (WEBQUESTIONS), **Small-Scale** subset of Freebase KG
- Matches previously reported accuracy without requiring lexicon, rules, POS taggers, parsers.

Yih, He, & Meek (2014)

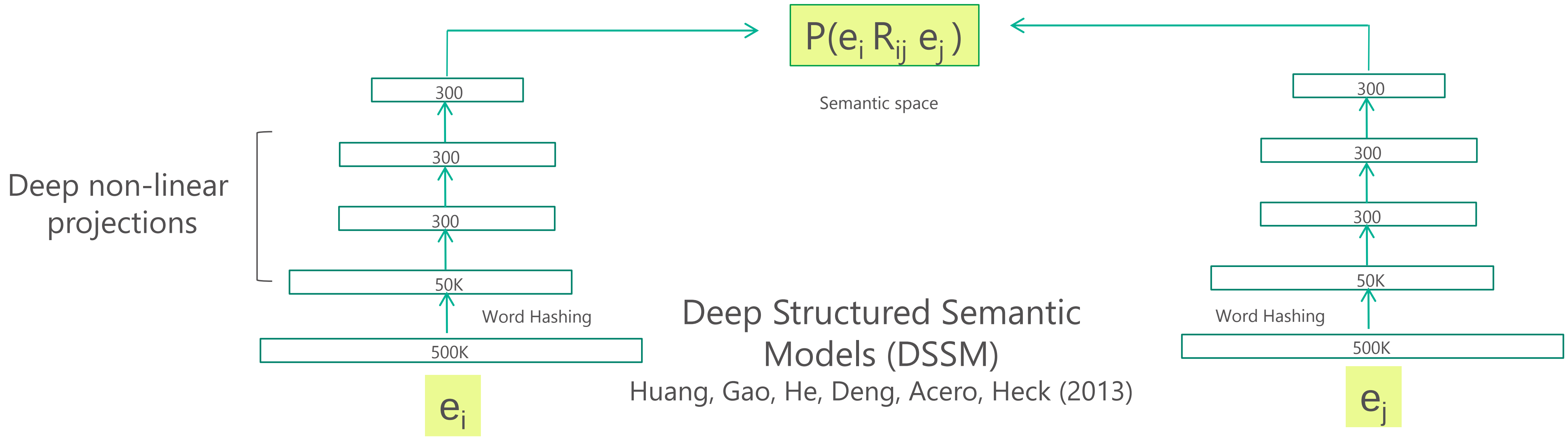
- DNN for single-relation QA
- **Supervised** training, **Small-Scale** KG tailored to PARALEX
- Achieves state-of-the-art performance (+7% F1)

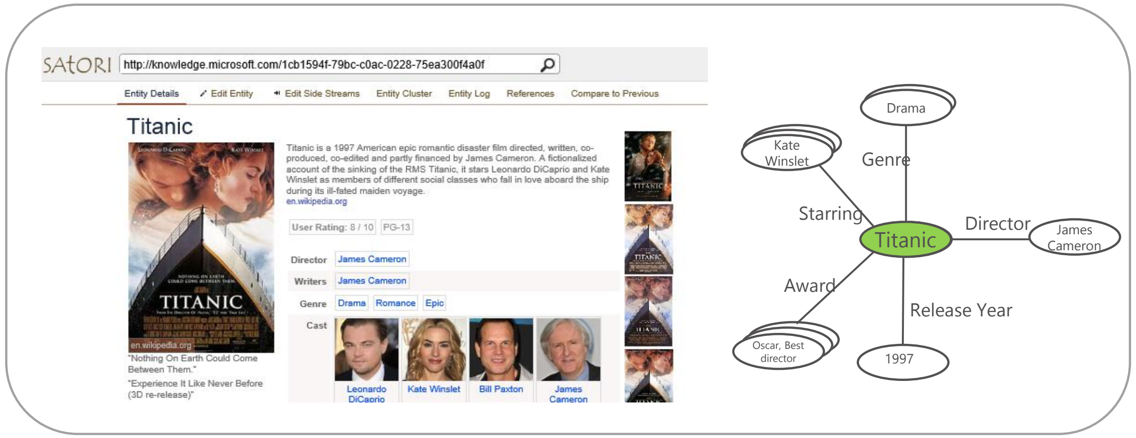
Huang & Heck (2014)

- Embedding model for open domain semantic parsing
- **Unsupervised** training (complete Wikipedia corpus), **Large-Scale** KG (Freebase)
- State-of-the-art entity disambiguation for Twitter Wikification task (26% error rate reduction)

Knowledge Embeddings (Huang & Heck 2014)

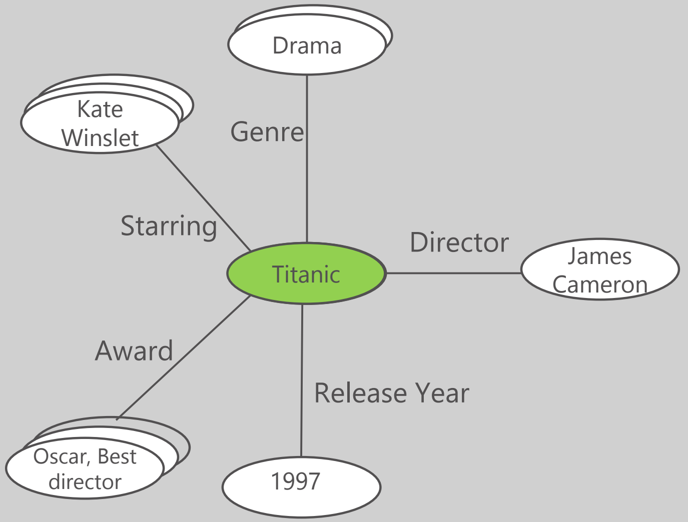
Deep Learning from Knowledge Graphs (DLKG) Huang and Heck, MSR-TR-2014-108, 2014





Knowledge Embeddings

Encoding the Input Features

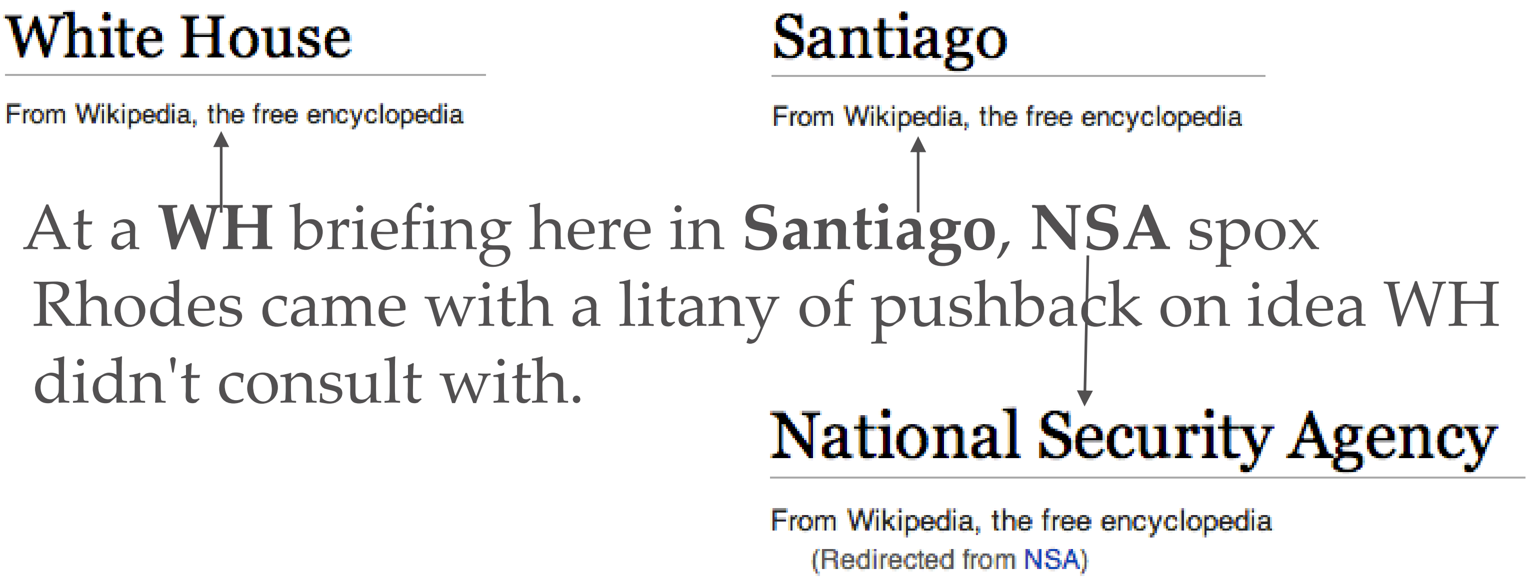
Knowledge	Representation	Example
Description	Letter tri-gram vector	dog = <#do, dog, og#> <0,…,1,1,…,0,1,…,0>
Entity Type	1-of-V vector	<0,….,0,…,1,…,0,…>
Subgraph	1-of-V vector for relation Letter tri-gram for entities	

Deep Learning from Knowledge Graphs

Experiment: Entity Disambiguation in Twitter

Disambiguate linkable mentions in Twitter tweets from a specific context to their referent entities in a Knowledge Graph

- A mention: a phrase referring to something in the world
- An entity: person, organization, object, event…



Deep Learning from Knowledge Graphs

Experiment: Data and Scoring Metric

Data

- A public data set includes 502 messages from 28 users (Meiji et al., 2012)
- A Wikipedia dump on May 3, 2013

Scoring Metric

- Accuracy on top ranked entity candidates

Baselines

- **TagMe**: **unsupervised** model based on prior popularity and semantic relatedness of single message (Ferragina and Scaiella, 2010)
- **Meij**: state-of-the-art **supervised** approach using the random forest model (Meij et al., 2012)

Deep Learning from Knowledge Graphs

Experiment: Results

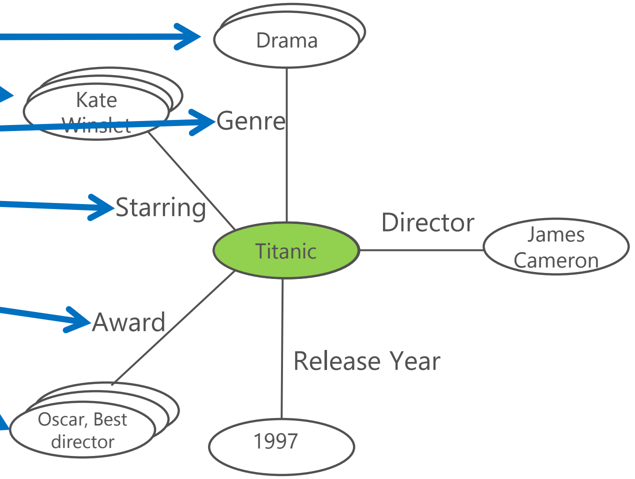
Method	Error Rate
TagMe (unsupervised)	38.1%
Meiji (supervised)	31.6%

Deep Learning from Knowledge Graphs

Experiment: Results

Unsupervised

Method	Error Rate
TagMe (unsupervised)	38.1%
Meiji (supervised)	31.6%
DLKG: Sub-graph (2nd Order Entities)	31.8%
+ Sub-graph (Relations)	30.0%
+ Entity Type (e.g., movie)	29.1%
+ Description (Wikipedia article)	**28.1%**



26.2% error rate reduction over TagMe (unsupervised, best reported accuracy)

For more details, see: Huang and Heck, MSR-TR-2014-108, 2014

Thank you!

For more questions, please email:

Dilek@ieee.org

Larry.Heck@ieee.org