Unsupervised Learning and Modeling of Knowledge and Intent for Spoken Dialogue Systems

YUN-NUNG (VIVIAN) CHEN HTTP://VIVIANCHEN.IDV.TW

APRIL 16TH, 2015 @ NEW YORK UNIVERSITY

Outline

Introduction

Ontology Induction [ASRU’13, SLT’14a]

Structure Learning [NAACL-HLT’15]

Semantic Decoding (submitted)

Conclusions

Outline

Introduction

Ontology Induction [ASRU'13, SLT'14a]

Structure Learning [NAACL-HLT'15]

Semantic Decoding (submitted)

Conclusions

A Popular Robot - Baymax

Big Hero 6 -- Video content owned and licensed by Disney Entertainment, Marvel Entertainment, LLC, etc

A Popular Robot - Baymax

Baymax is capable of maintaining a good **spoken dialogue system** and **learning** new knowledge for better **understanding** and **interacting** with people.

The goal is to automate learning and understanding procedures in system development.

Spoken Dialogue System (SDS)

Spoken dialogue systems are the intelligent agents that are able to help users finish tasks more efficiently via speech interactions.

Spoken dialogue systems are being incorporated into various devices (smart-phones, smart TVs, in-car navigating system, etc).

Apple's Siri

Microsoft's Cortana

Amazon's Echo

Samsung's SMART TV

Google Now

https://www.apple.com/ios/siri/
http://www.windowsphone.com/en-us/how-to/wp8/cortana/meet-cortana
http://www.amazon.com/oc/echo/
http://www.samsung.com/us/experience/smart-tv/
https://www.google.com/landing/now/

Large Smart Device Population

The number of global smartphone users will surpass 2 billion in 2016.

As of 2012, there are 1.1 billion automobiles on the earth.

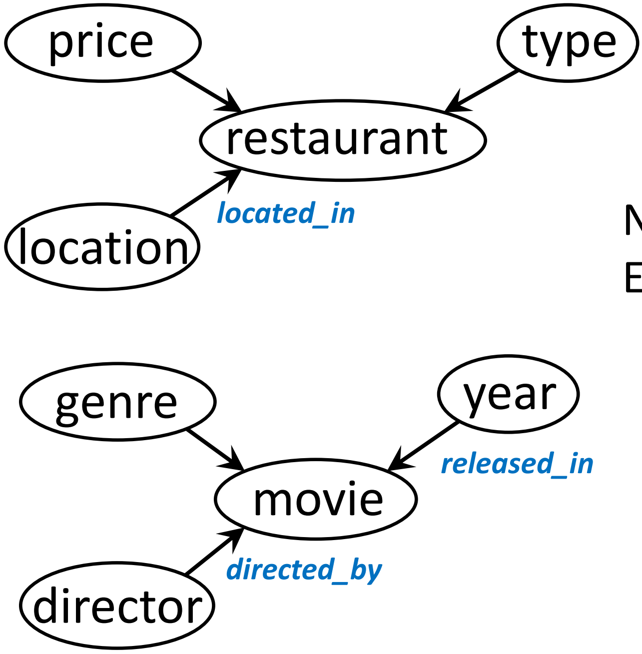
The more **natural** and **convenient** input of the devices evolves towards speech

Knowledge Representation/Ontology

Traditional SDSs require **manual annotations** for **specific domains** to represent domain knowledge.

Restaurant Domain

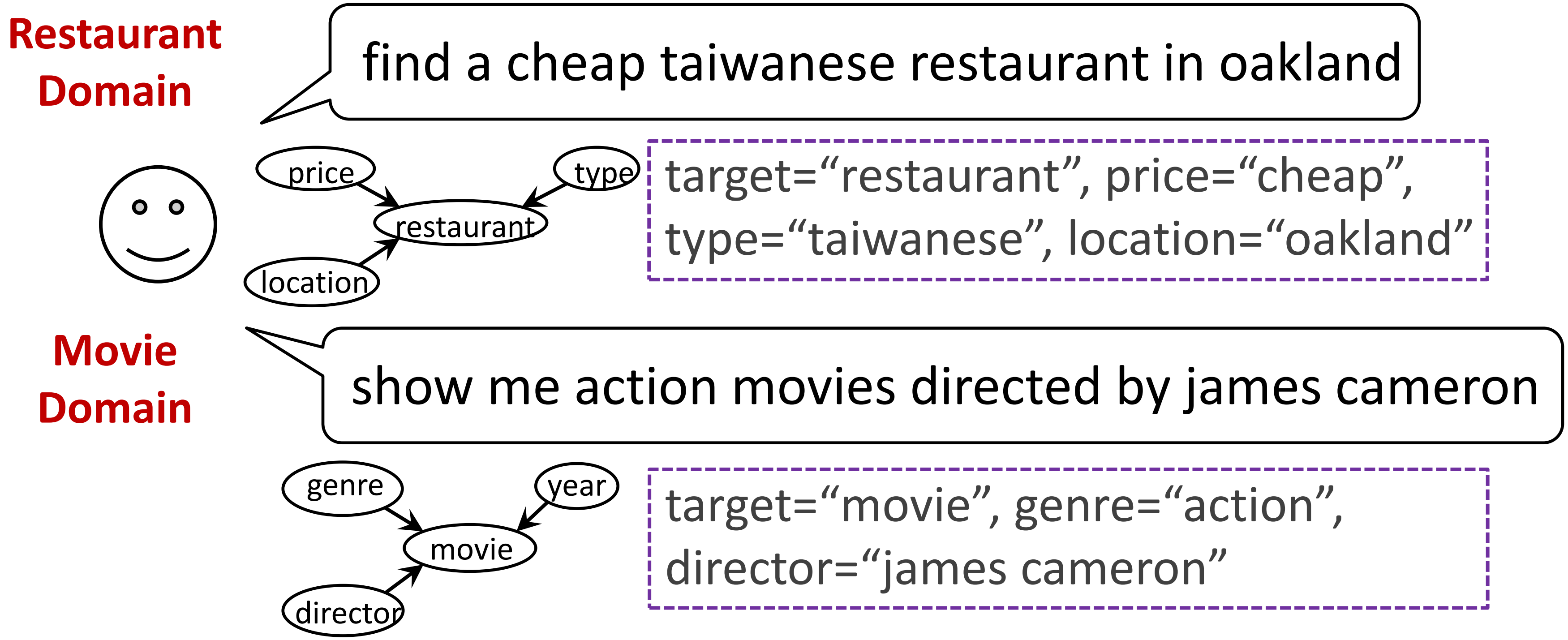
Movie Domain



Node: semantic concept/slot
Edge: relation between concepts

Utterance Semantic Representation

A spoken language understanding (SLU) component requires the domain ontology to decode utterances into semantic forms, which contain **core content (a set of slots and slot-fillers)** of the utterance.



Challenges for SDS

An SDS in a new domain requires

1) A hand-crafted domain ontology
2) Utterances labelled with semantic representations
3) An SLU component for mapping utterances into semantic representations

With increasing spoken interactions, building domain ontologies and annotating utterances cost a lot so that the data does not scale up.

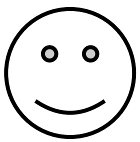
The goal is to enable an SDS to automatically learn this knowledge so that open domain requests can be handled.

Questions to Address

1) Given unlabelled raw audio recordings, how can a system automatically induce and organize domain-specific concepts?
2) With the automatically acquired knowledge, how can a system understand individual utterances?

Interaction Example

User



find a cheap restaurant for asian food

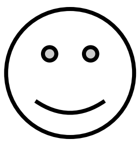
Cheap Asian restaurants include Kelly & Ping,
Saigon Shack, etc.
What do you want to choose?

Intelligent Agent

Q: How does a dialogue system process this request?

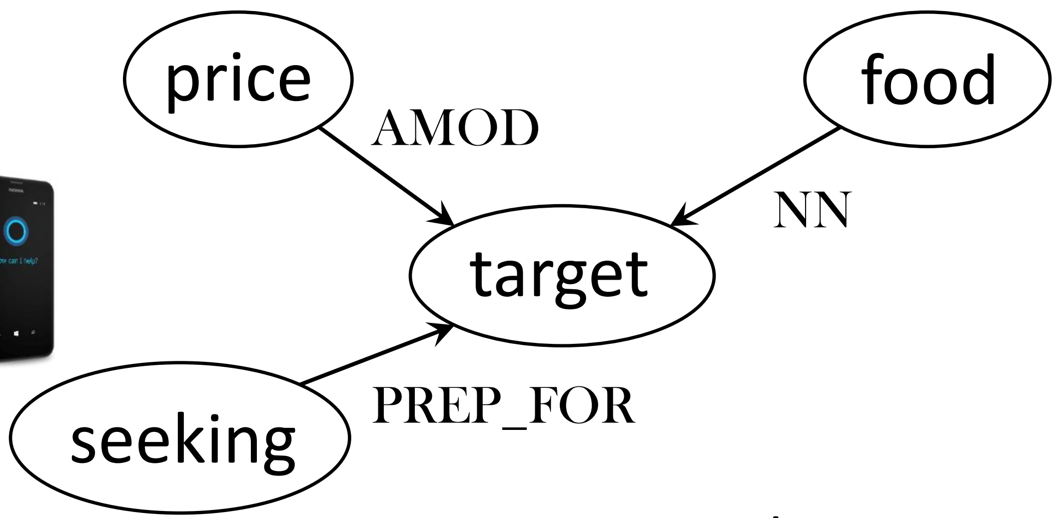
SDS Process – Available Domain Ontology

User



find a cheap restaurant for asian food

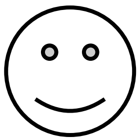
Intelligent Agent



Organized Domain Knowledge

SDS Process – Available Domain Ontology

User

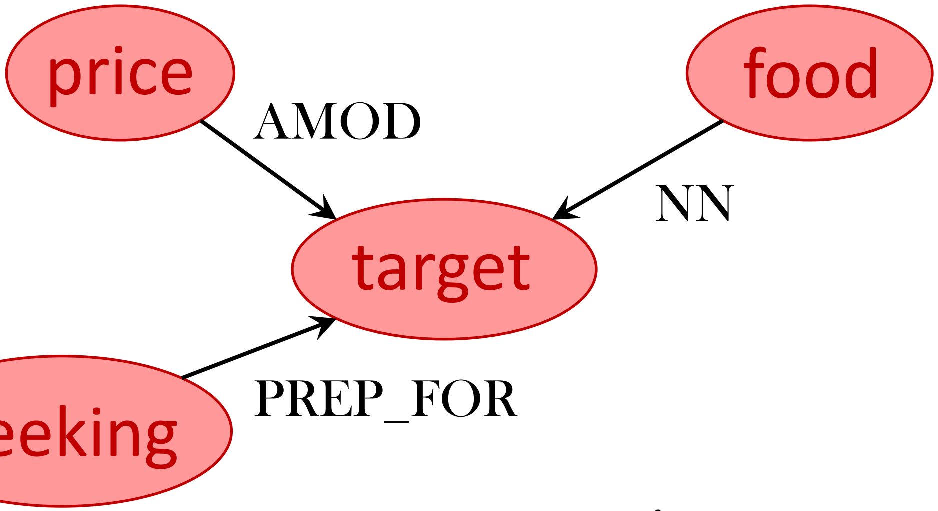


find a cheap restaurant for asian food

Ontology Induction

(semantic slot)

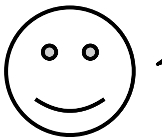
Intelligent Agent



Organized Domain Knowledge

SDS Process – Available Domain Ontology

User

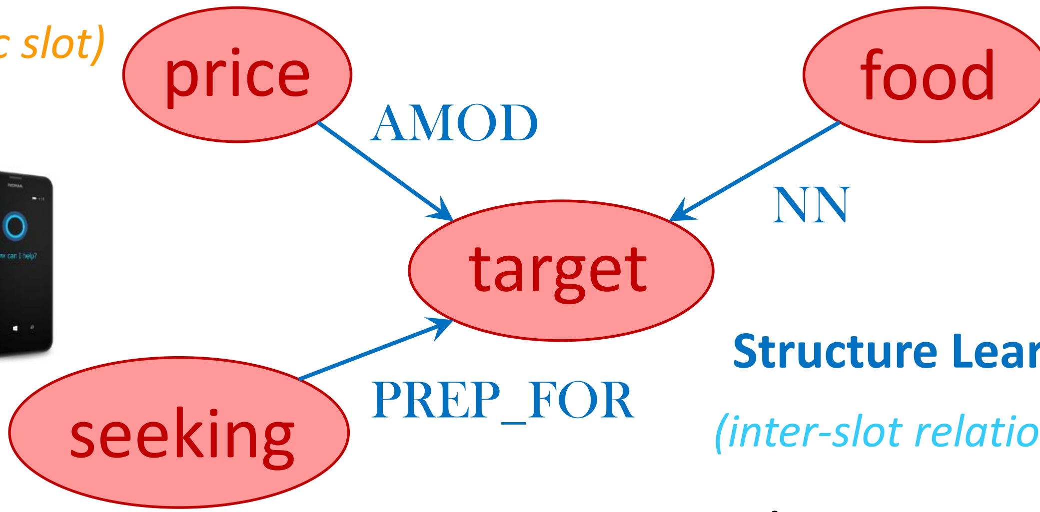


find a cheap restaurant for asian food

Ontology Induction

(semantic slot)

Intelligent Agent

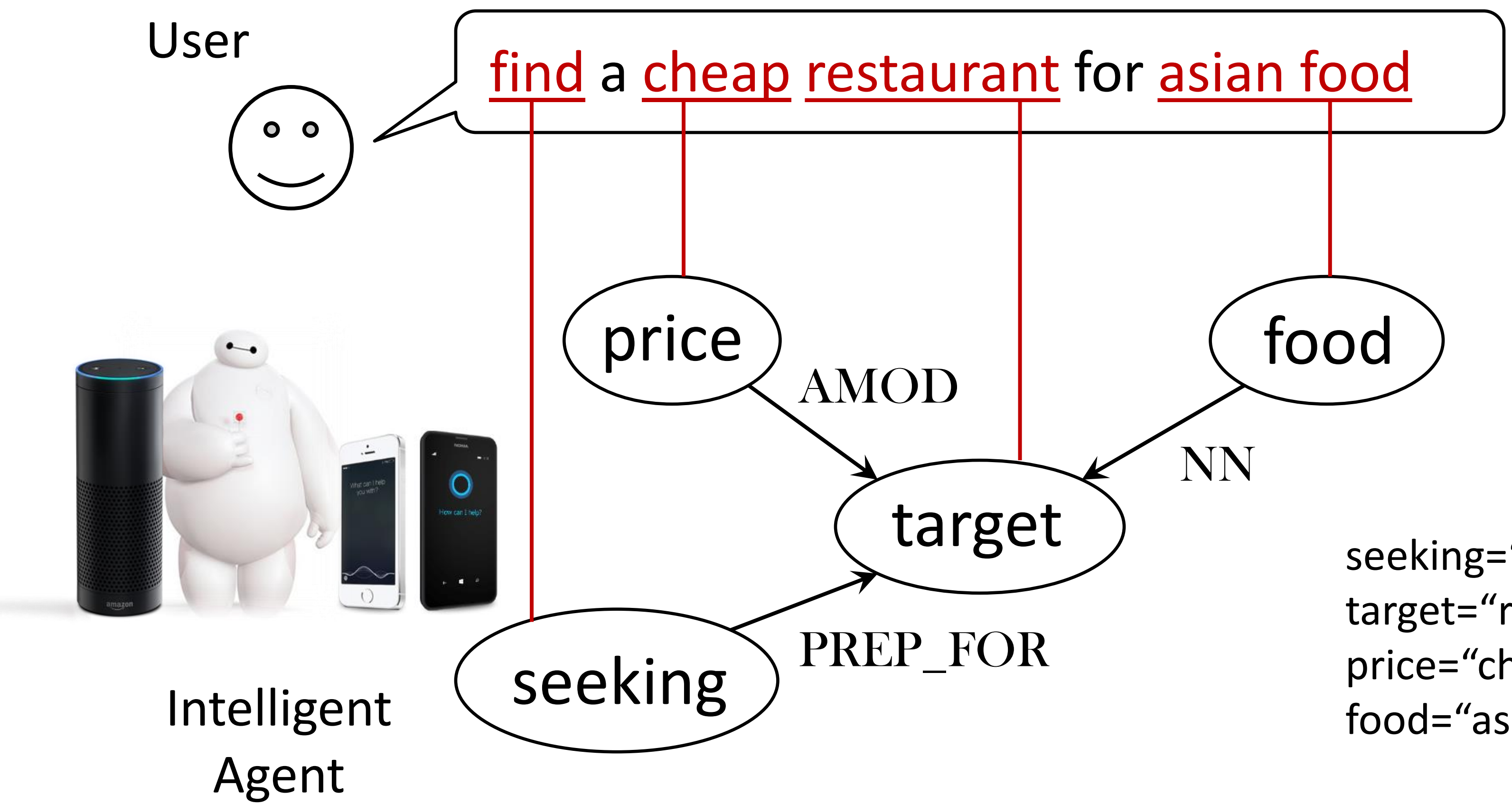


Structure Learning

(inter-slot relation)

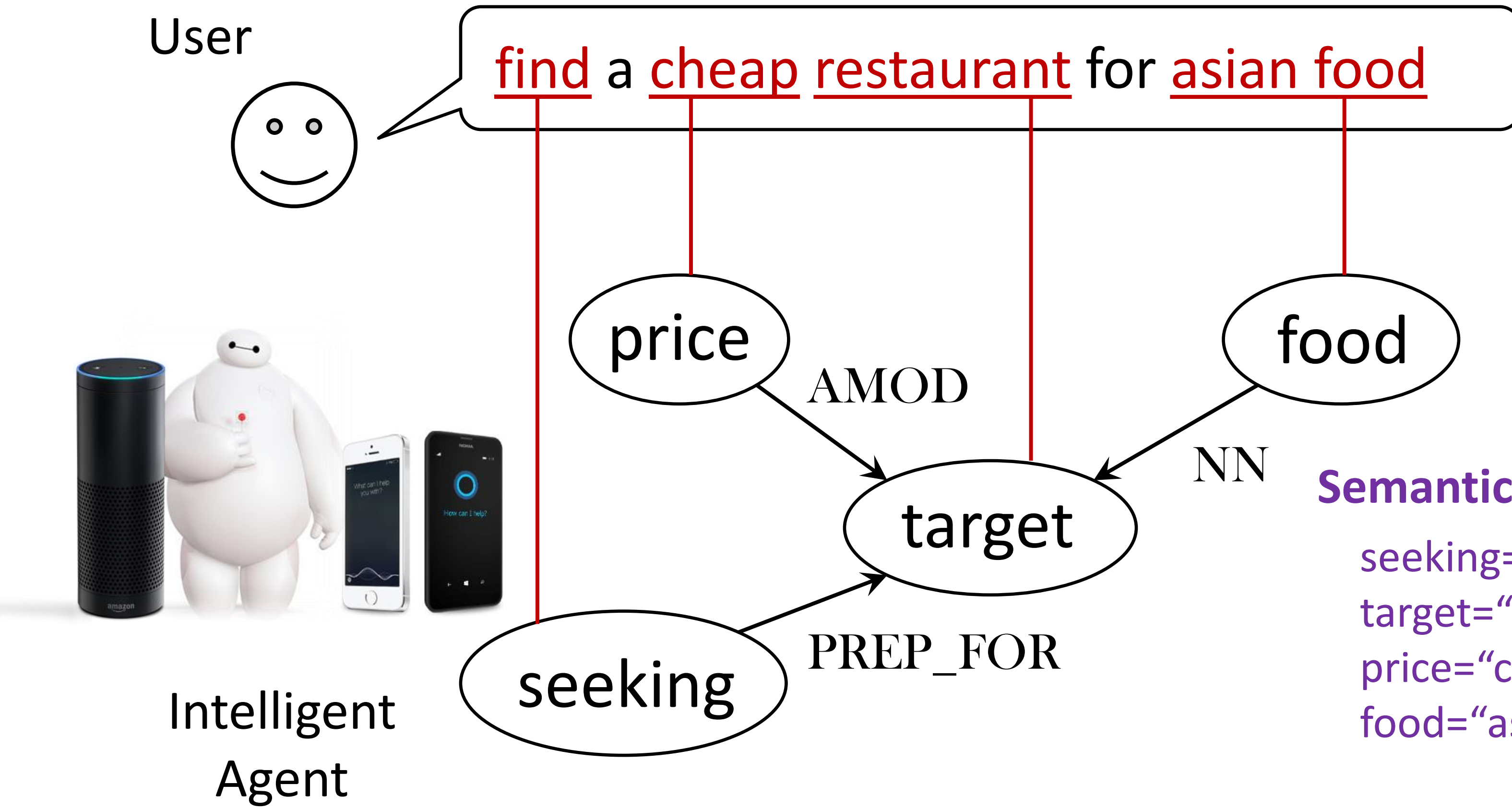
Organized Domain Knowledge

SDS Process – Spoken Language Understanding (SLU)



seeking="find"
target="restaurant"
price="cheap"
food="asian food"

SDS Process – Spoken Language Understanding (SLU)

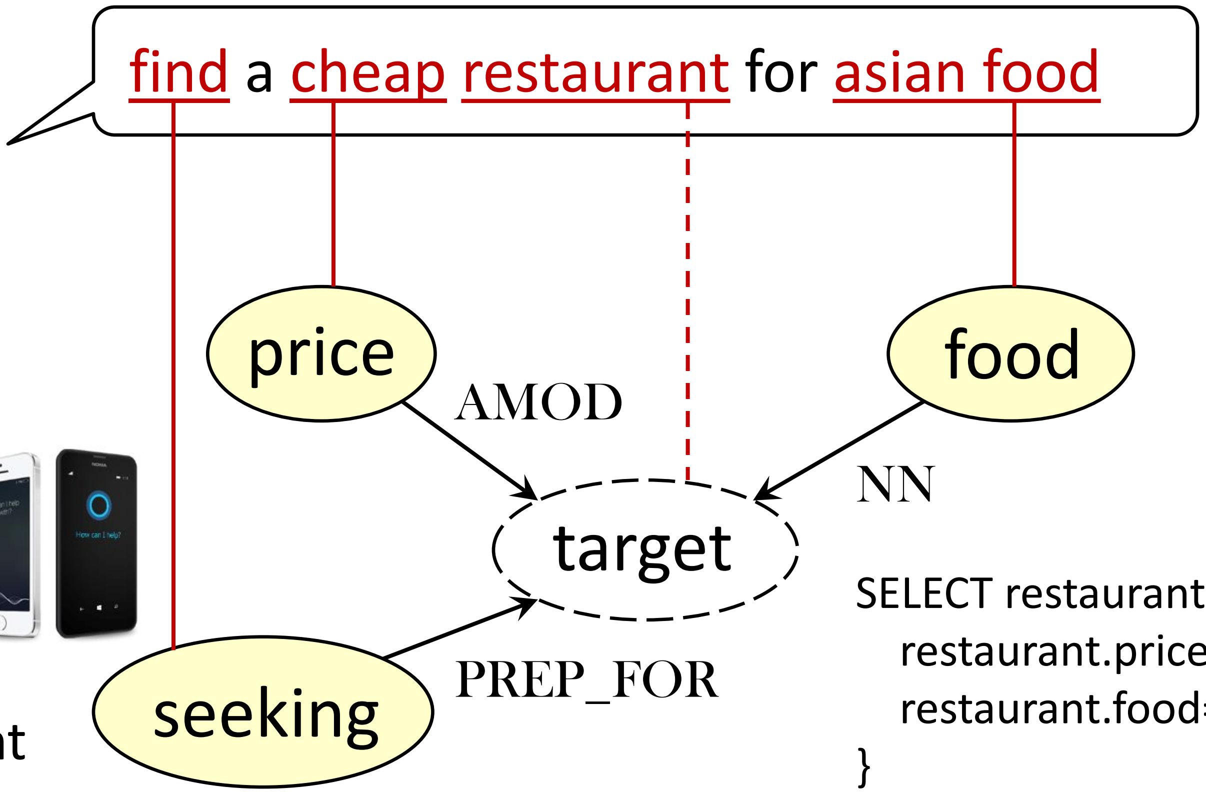


Semantic Decoding

seking="find"
target="restaurant"
price="cheap"
food="asian food"

SDS Process – Dialogue Management (DM)

User

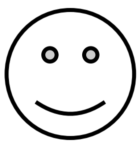


Intelligent Agent

```
SELECT restaurant {
    restaurant.price="cheap"
    restaurant.food="asian food"
}
```

SDS Process – Dialogue Management (DM)

User



find a cheap restaurant for asian food

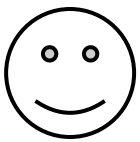
```
SELECT restaurant {
    restaurant.price="cheap"
    restaurant.food="asian food"
}
```

Kelly & Ping
Saigon Shack
:
:

Intelligent
Agent

SDS Process – Natural Language Generation (NLG)

User



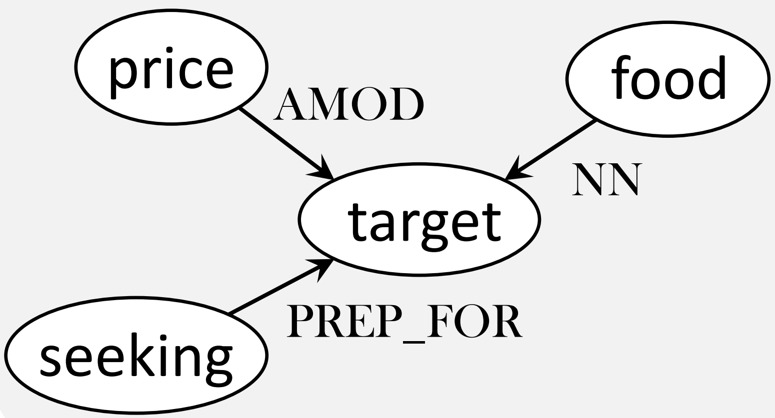
find a cheap restaurant for asian food

Intelligent Agent

Cheap Asian restaurants include Kelly & Ping, Saigon Shack, etc.
What do you want to choose?

Goals

User

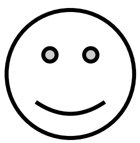



```
SELECT restaurant {
    restaurant.price="cheap"
    restaurant.food="asian food"
}
```

Required Domain-Specific Information

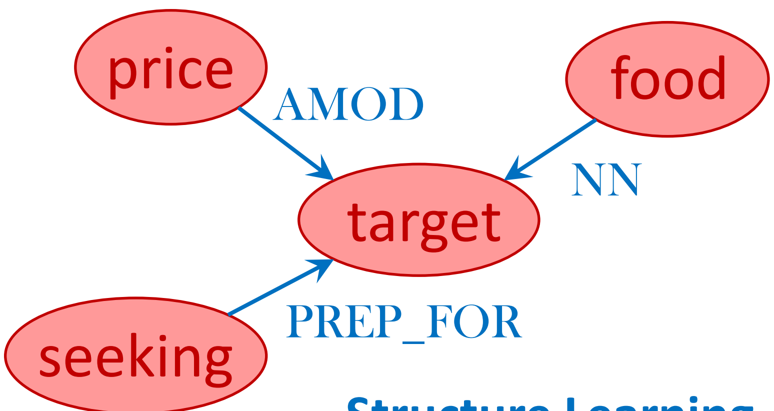
Goals

User



find a cheap restaurant for asian food

Ontology Induction *(semantic slot)*



Structure Learning

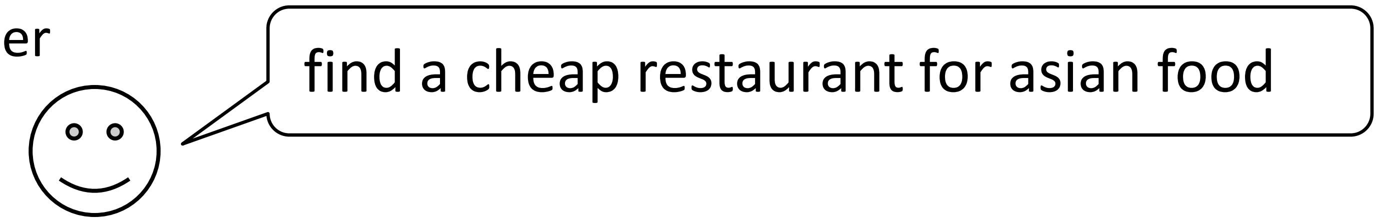
(inter-slot relation)

SELECT restaurant {
restaurant.price="cheap"
restaurant.food="asian food"
}

Semantic Decoding

Goals

User



Ontology Induction

Semantic Decoding

Structure Learning

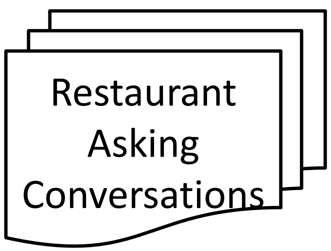
Knowledge Acquisition

1) Given unlabelled raw audio recordings, how can a system automatically induce and organize domain-specific concepts?

Knowledge Acquisition

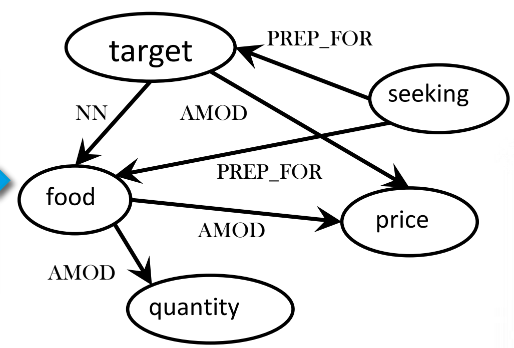
Ontology Induction

Structure Learning



Unlabelled Collection

Knowledge Acquisition

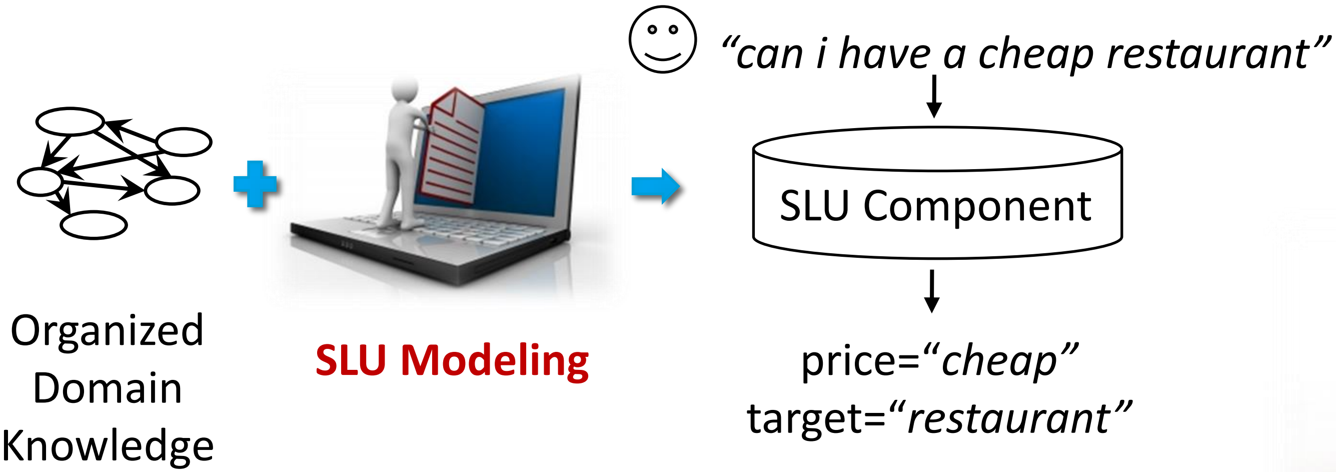


Organized Domain Knowledge

SLU Modeling

2) With the automatically acquired knowledge, how can a system understand individual utterances?

SLU Modeling **Semantic Decoding**



Outline

Introduction

Ontology Induction [ASRU'13, SLT'14a]

Structure Learning [NAACL-HLT'15]

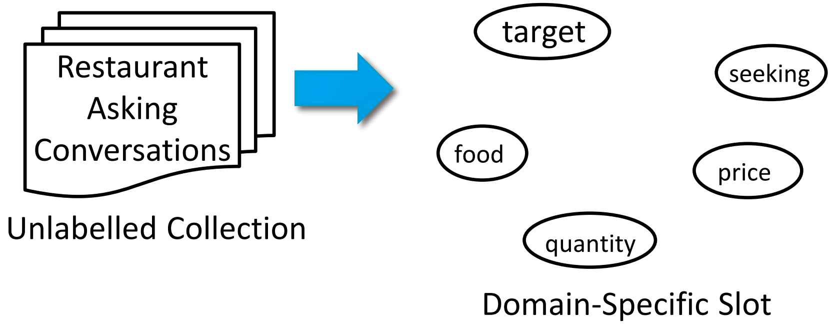
Semantic Decoding (submitted)

Conclusions

Ontology Induction [ASRU'13, SLT'14a]

Input: Unlabelled user utterances

Output: Slots that are useful for a domain-specific SDS



Step 1: Frame-semantic parsing on all utterances for creating slot candidates

Step 2: Slot ranking model for differentiating domain-specific concepts from generic concepts

Step 3: Slot selection

Y.-N. Chen et al., "Unsupervised Induction and Filling of Semantic Slots for Spoken Dialogue Systems Using Frame-Semantic Parsing," in *Proc. of ASRU,* 2013. **(Best Student Paper Award)**
Y.-N. Chen et al., "Leveraging Frame Semantics and Distributional Semantics for Unsupervised Semantic Slot Induction in Spoken Dialogue Systems," in *Proc. of SLT,* 2014.

Probabilistic Frame-Semantic Parsing

FrameNet [Baker et al., 1998]

- a linguistically semantic resource, based on the frame-semantics theory
- “low fat milk” → “milk” evokes the “food” frame;
  “low fat” fills the descriptor frame element

SEMAFOR [Das et al., 2014]

- a state-of-the-art frame-semantics parser, trained on manually annotated FrameNet sentences

Baker et al., " The berkeley framenet project," in Proc. of International Conference on Computational linguistics, 1998.
Das et al., " Frame-semantic parsing," in Proc. of Computational Linguistics, 2014.

Step 1: Frame-Semantic Parsing for Utterances

Good!

Good!

?

can i have a cheap restaurant

Frame: capability
FT LU: can FE LU: i

Frame: expensiveness
FT LU: cheap

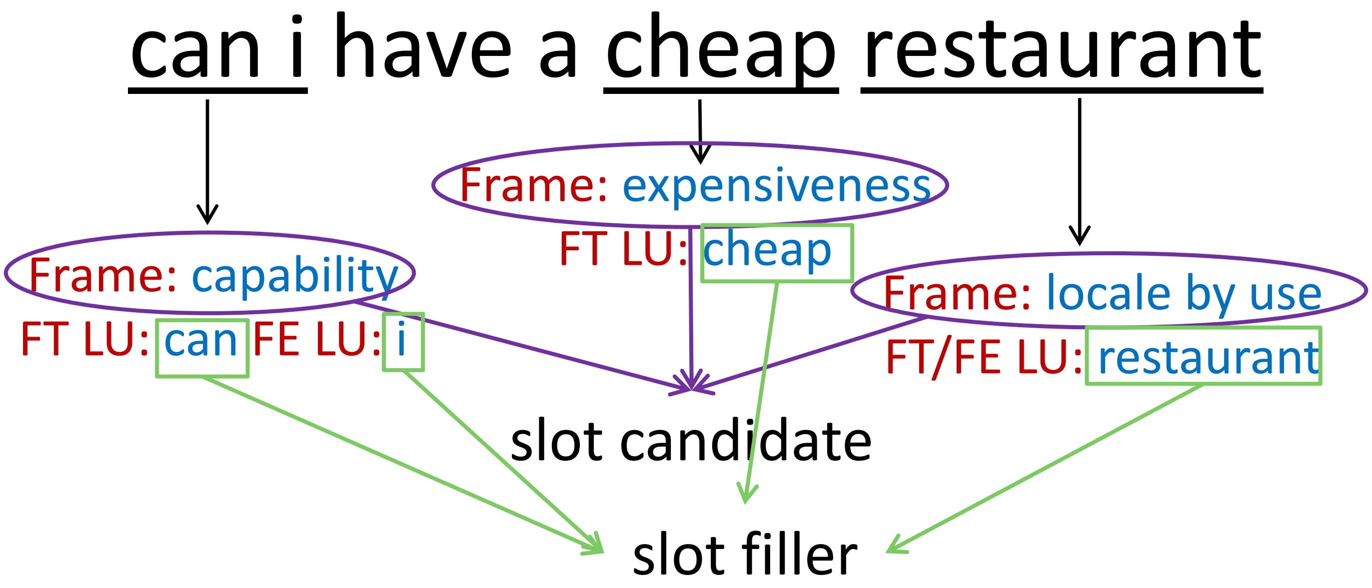
Frame: locale by use
FT/FE LU: restaurant

FT: Frame Target; FE: Frame Element; LU: Lexical Unit

Task: adapting ***generic*** frames to ***domain-specific*** settings for SDSs

Step 2: Slot Ranking Model

Main Idea: rank *domain-specific* concepts higher than *generic* semantic concepts



Step 2: Slot Ranking Model

Rank a slot candidate s by integrating two scores

$$w(s) = (1-\alpha)\log \underbrace{f(s)} + \alpha \cdot \log \underbrace{h(s)}$$

slot frequency in the domain-specific conversation

slots with higher frequency → more important

semantic coherence of slot fillers

domain-specific concepts → fewer topics

Step 2: Slot Ranking Model

$h(s)$: Semantic coherence

slot name: quantity

a one three all

lower coherence in topic space

slot name: expensiveness

cheap inexpensive expensive

higher coherence in topic space

$$h(s) = \frac{1}{\boxed{|V(s)|}^2} \sum_{x_a, x_b \in V(s)} \boxed{\mathrm{Sim}(x_a, x_b)}$$

the slot-filler set corresponding to s

measured by cosine similarity between their word embeddings

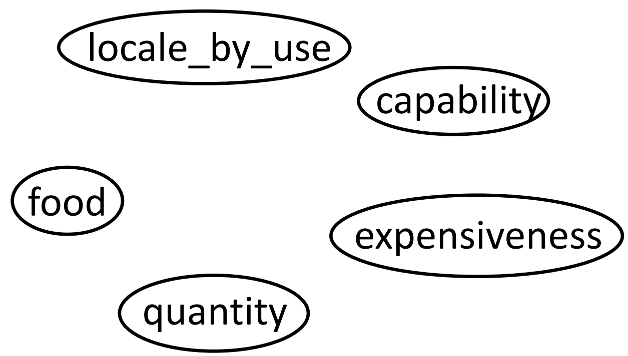
Step 3: Slot Selection

Rank all slot candidates by their importance scores

$$w(s) = (1-\alpha)\log \underbrace{f(s)}_{\text{frequency}} + \alpha \cdot \log \underbrace{h(s)}_{\text{semantic coherence}}$$

Output slot candidates with higher scores based on a threshold

locale_by_use	0.89
capability	0.68
food	0.64
expensiveness	0.49
quantity	0.30
seeking	0.22
:	



Experiments of Ontology Induction

Dataset
- Cambridge University SLU corpus [Henderson, 2012]
  - Restaurant recommendation in an in-car setting in Cambridge
    - WER = 37%
    - vocabulary size = 1868
    - 2,166 dialogues
    - 15,453 utterances
    - dialogue slot: addr, area, food, name, phone, postcode, price range, task, type

Induced slots	Reference slot
locale by use, building	type
food, origin	food
speak on topic	addr
seeking, desiring, locating	task
commerce scenario, expensiveness, range	price range
part orientational, direction, locale, part inner outer	area
contacting	phone
sending	postcode

The mapping table between induced and reference slots

Henderson et al., "Discriminative spoken language understanding using word confusion networks," in Proc. of SLT, 2012.

Experiments of Ontology Induction

- **Slot Induction Evaluation**: Average Precision (AP) and Area Under the Precision-Recall Curve (AUC) of the slot ranking model to measure quality of induced slots via the mapping table

Approach	ASR		Manual	
	AP (%)	AUC (%)	AP (%)	AUC (%)
Baseline: MLE	56.7	54.7	53.0	50.8
MLE + Semantic Coherence	**71.7** (+26.5%)	**70.4** (+28.7%)	**74.4** (+40.4%)	**73.6** (+44.9%)

Semantic relations help decide domain-specific knowledge.

Induced slots have 70% of AP and align well with human-annotated slots for SDS.

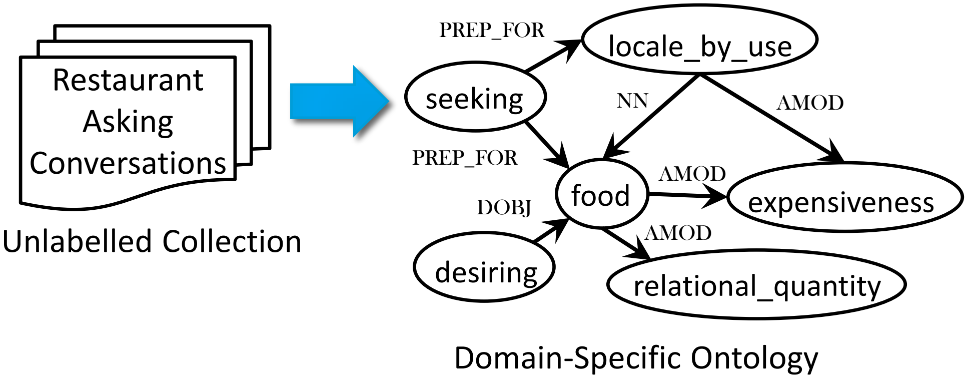
Outline

- Introduction
- Ontology Induction [ASRU'13, SLT'14a]
- **Structure Learning [NAACL-HLT'15]**
- Semantic Decoding (submitted)
- Conclusions

Structure Learning [NAACL-HLT'15]

Input: Unlabelled user utterances

Output: Slots <u>with relations</u>



Step 1: Construct a graph to represent slots, words, and relations

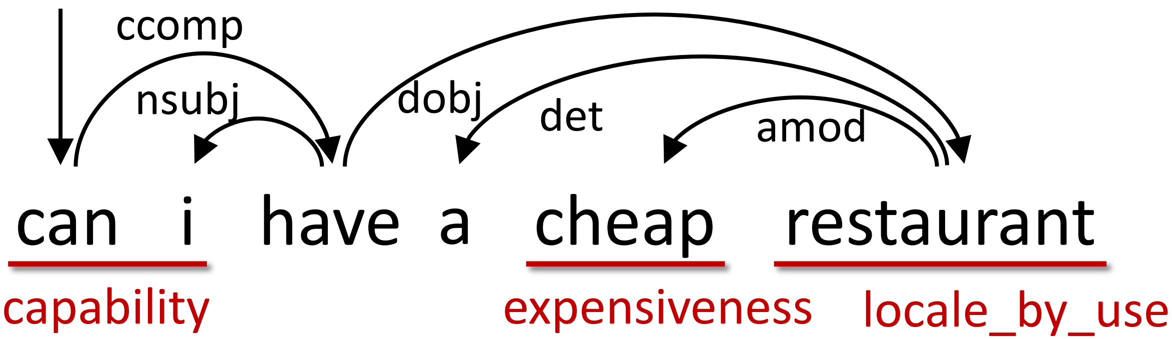
Step 2: Compute scores for edges (relations) and nodes (slots)

Step 3: Identify important relations connecting important slot pairs

<u>Y.-N. Chen</u> et al., "Jointly Modeling Inter-Slot Relations by Random Walk on Knowledge Graphs for Unsupervised Spoken Language Understanding," *in Proc. of NAACL-HLT,* 2015.

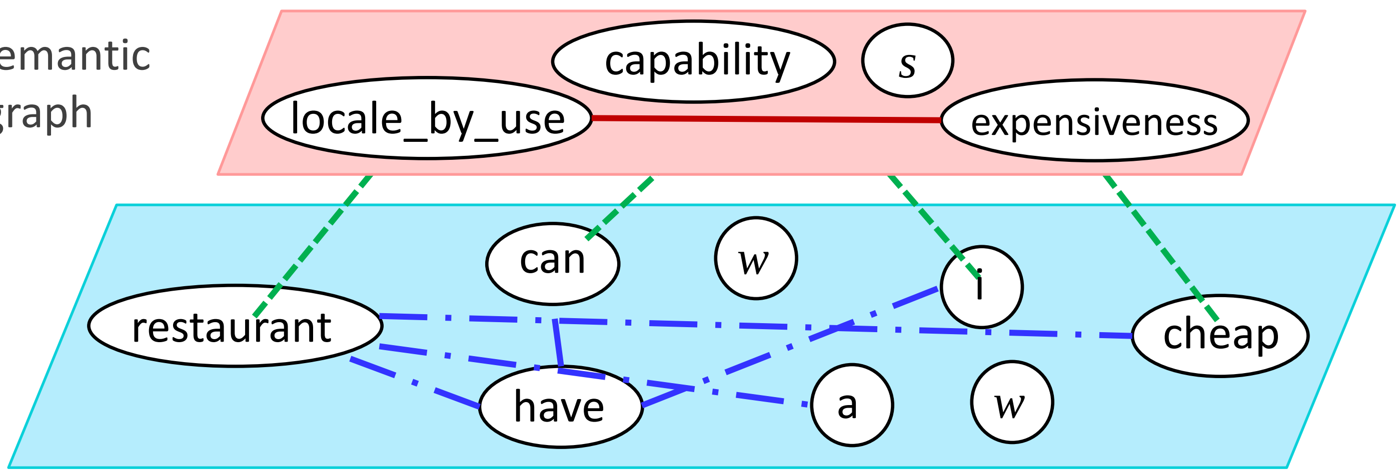
Step 1: Knowledge Graph Construction

Syntactic dependency parsing on utterances



Slot-based semantic knowledge graph

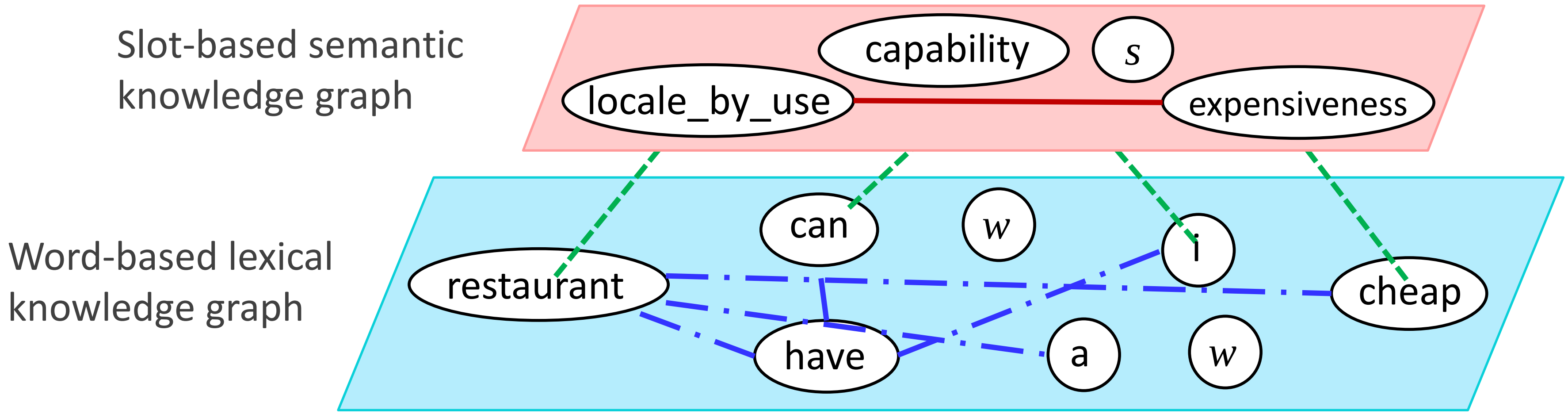
Word-based lexical knowledge graph



Step 1: Knowledge Graph Construction

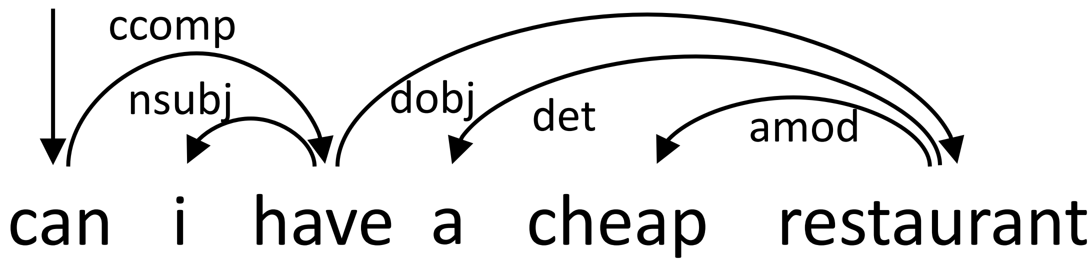
The edge between a node pair is weighted as relation importance

How to decide the weights to represent relation importance?



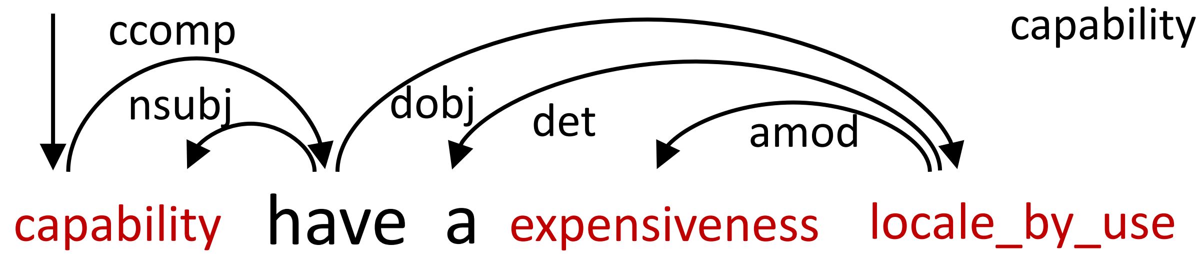
Step 2: Weight Measurement
Slot/Word Embeddings Training

Dependency-based word embeddings



$\text{can} = [0.8 \quad \ldots \quad 0.24]$
$\text{have} = [0.3 \quad \ldots \quad 0.21]$
:
:

Dependency-based slot embeddings



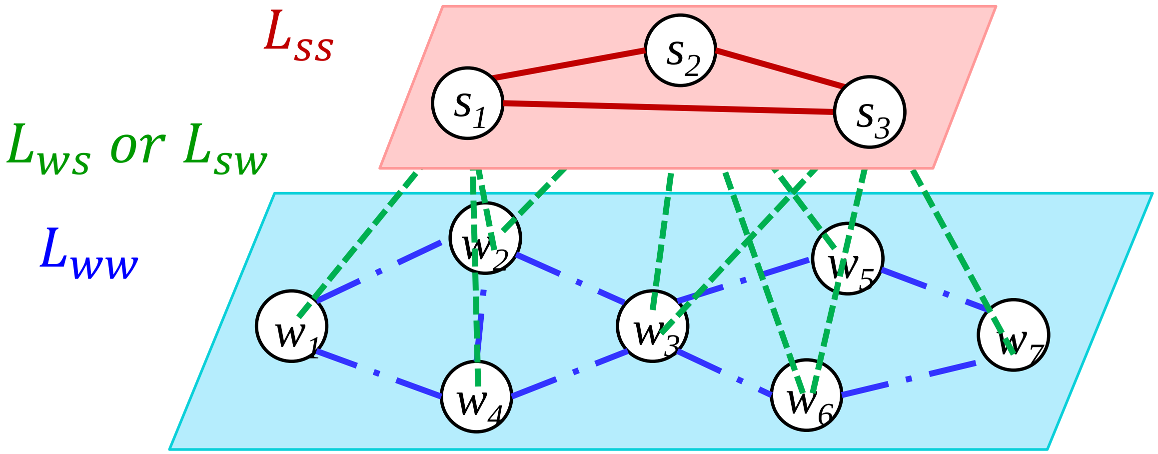
$\text{expensiveness} = [0.12 \quad \ldots \quad 0.7]$
$\text{capability} = [0.3 \quad \ldots \quad 0.6]$
:
:

Levy and Goldberg, " Dependency-Based Word Embeddings," in *Proc. of ACL*, 2014.

Step 2: Weight Measurement

Compute edge weights to represent relation importance

- Slot-to-slot relation L_{ss}: similarity between slot embeddings
- Word-to-slot relation L_{ws} or L_{sw}: frequency of the slot-word pair
- Word-to-word relation L_{ww}: similarity between word embeddings



Step 2: Slot Importance by Random Walk

Assumption: the slots with more dependencies to more important slots should be more important

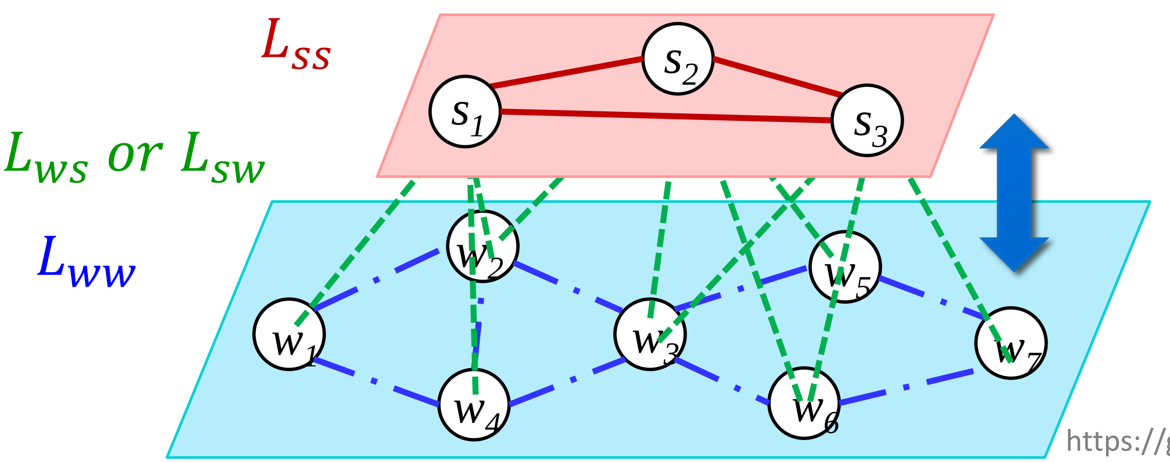
The random walk algorithm computes importance for each slot

slot importance

$$\begin{cases} \boxed{r_s^{(t+1)}} = (1-\alpha)\boxed{r_s^{(0)}} + \alpha \boxed{L_{ss} L_{sw} r_w^{(t)}} \\ r_w^{(t+1)} = (1-\alpha) r_w^{(0)} + \alpha L_{ww} L_{ws} r_s^{(t)} \end{cases}$$

original frequency score

scores propagated from word-layer then propagated within slot-layer



Converged scores can be obtained by a closed form solution.

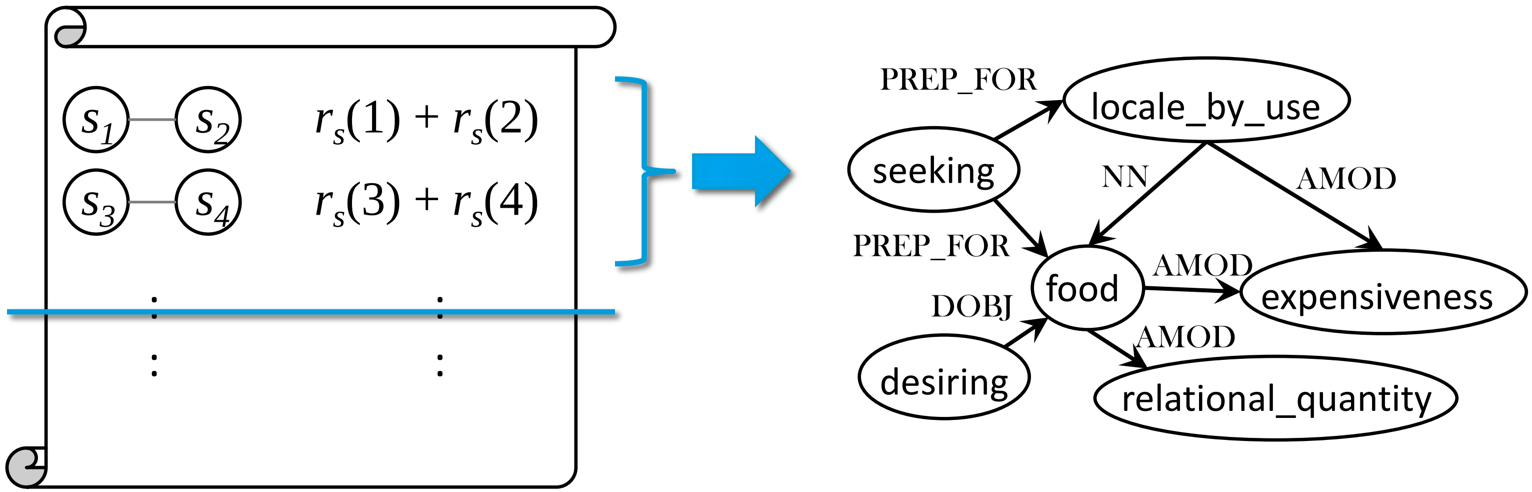
https://github.com/yvchen/MRRW

Step 3: Identify Domain Slots w/ Relations

The converged slot importance suggests whether the slot is important *(Experiment 1)*

Rank slot pairs by summing up their converged slot importance

Select slot pairs with higher scores according to a threshold *(Experiment 2)*



Experiments for Structure Learning

Experiment 1: Quality of Slot Importance

Dataset: Cambridge University SLU Corpus

Approach	ASR		Manual	
	AP (%)	AUC (%)	AP (%)	AUC (%)
Baseline: MLE	56.7	54.7	53.0	50.8
Random Walk: MLE + Dependent Relations	**69.0** (+21.8%)	**68.5** (+24.8%)	**75.2** (+41.8%)	**74.5** (+46.7%)

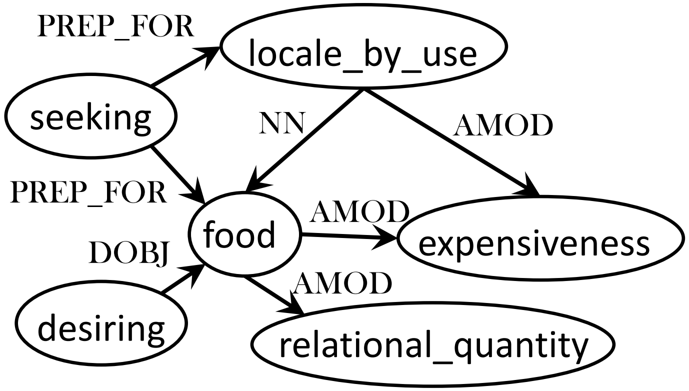
Dependent relations help decide domain-specific knowledge.

Experiments for Structure Learning

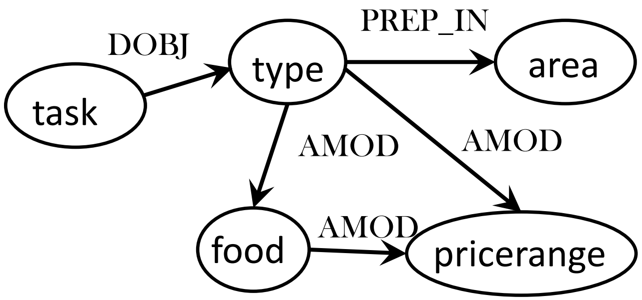
Experiment 2: Relation Discovery Evaluation

Discover inter-slot relations connecting important slot pairs

Rank	Relation
1	⟨locale_by_use, NN, food⟩
2	⟨food, AMOD, expensiveness⟩
3	⟨locale_by_use, AMOD, expensiveness⟩
4	⟨seeking, PREP_FOR, food⟩
5	⟨food, AMOD, relational_quantity⟩
6	⟨desiring, DOBJ, food⟩
7	⟨seeking, PREP_FOR, locale_by_use⟩
8	⟨food, DET, quantity⟩



The reference ontology with the most frequent syntactic dependencies

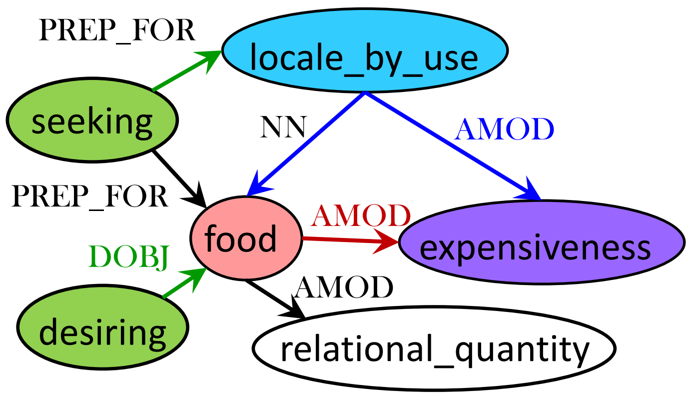


Experiments for Structure Learning

Experiment 2: Relation Discovery Evaluation

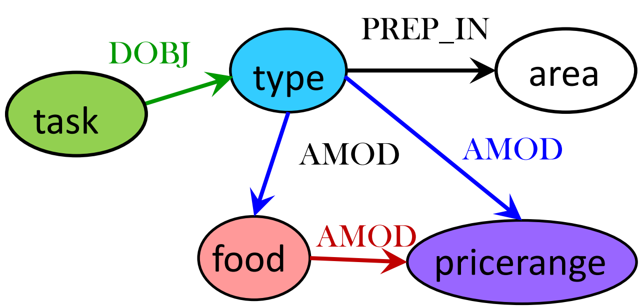
Discover inter-slot relations connecting important slot pairs

Rank	Relation
1	⟨locale_by_use, NN, food⟩
2	⟨food, AMOD, expensiveness⟩
3	⟨locale_by_use, AMOD, expensiveness⟩
4	⟨seeking, PREP_FOR, food⟩
5	⟨food, AMOD, relational_quantity⟩
6	⟨desiring, DOBJ, food⟩
7	⟨seeking, PREP_FOR, locale_by_use⟩
8	⟨food, DET, quantity⟩



The reference ontology with the most frequent syntactic dependencies

The automatically learned domain ontology aligns well with the reference one.



Outline

- Introduction
- Ontology Induction [ASRU'13, SLT'14a]
- Structure Learning [NAACL-HLT'15]
- Semantic Decoding (submitted)
- Conclusions

Knowledge Acquisition

Outline

Introduction

Ontology Induction [ASRU'13, SLT'14a]

Structure Learning [NAACL-HLT'15]

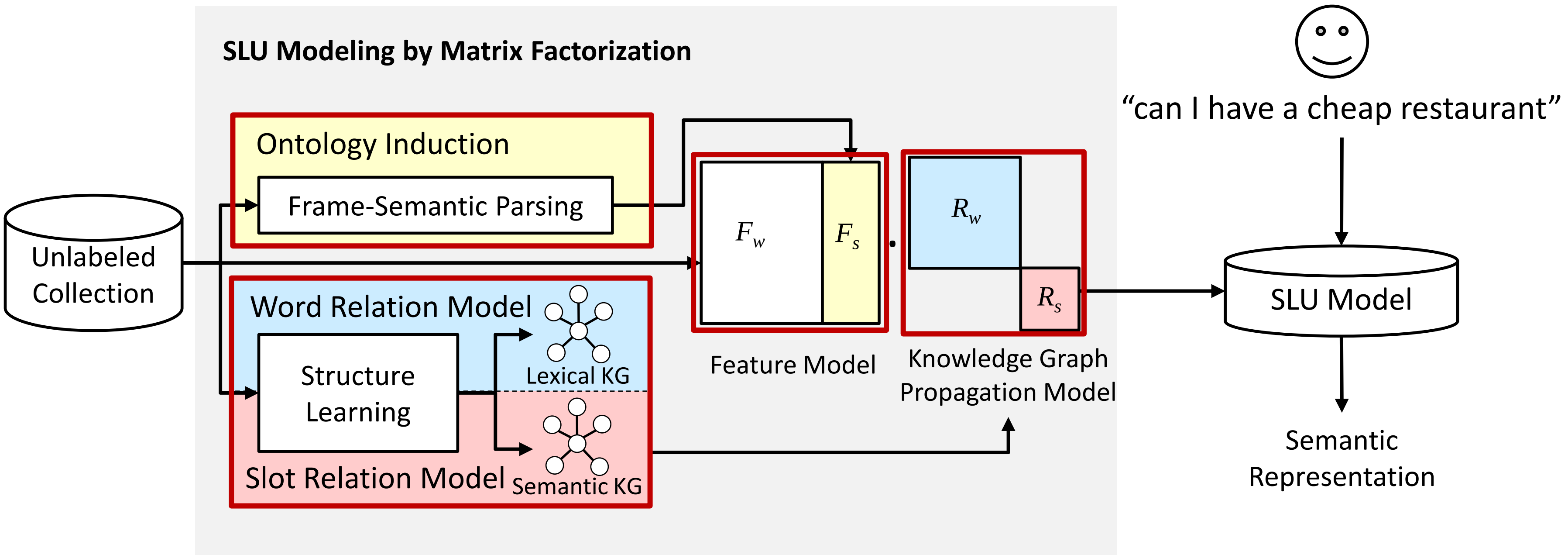
Semantic Decoding (submitted)

Conclusions

Semantic Decoding

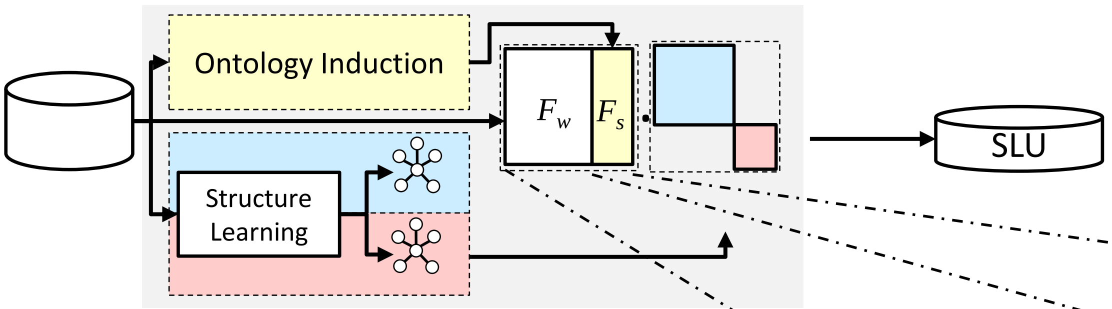
Input: user utterances, automatically learned knowledge

Output: the semantic concepts included in each individual utterance



Y.-N. Chen et al., "Matrix Factorization with Knowledge Graph Propagation for Unsupervised Spoken Language Understanding," submitted.

Matrix Factorization (MF)
Feature Model



Utterance 1

i would like a cheap restaurant

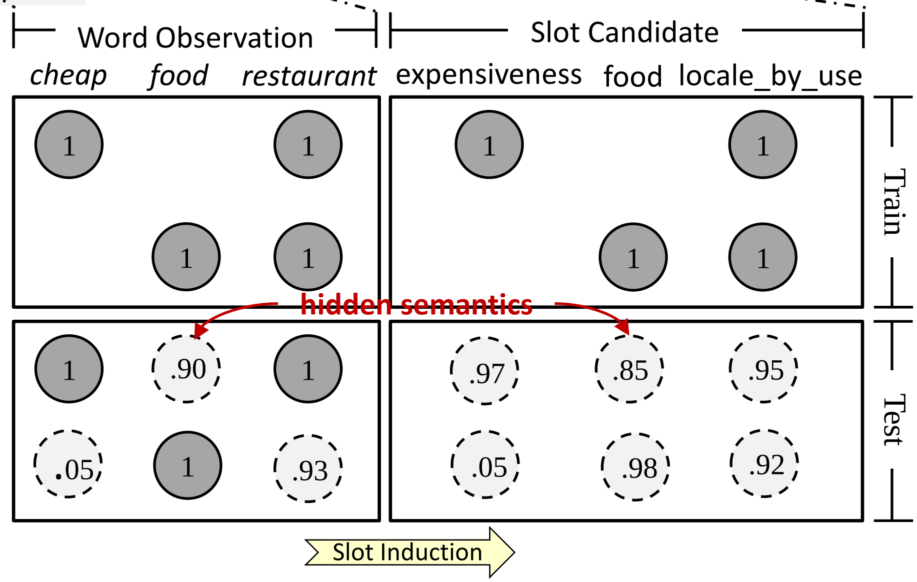
Utterance 2

find a restaurant with chinese food

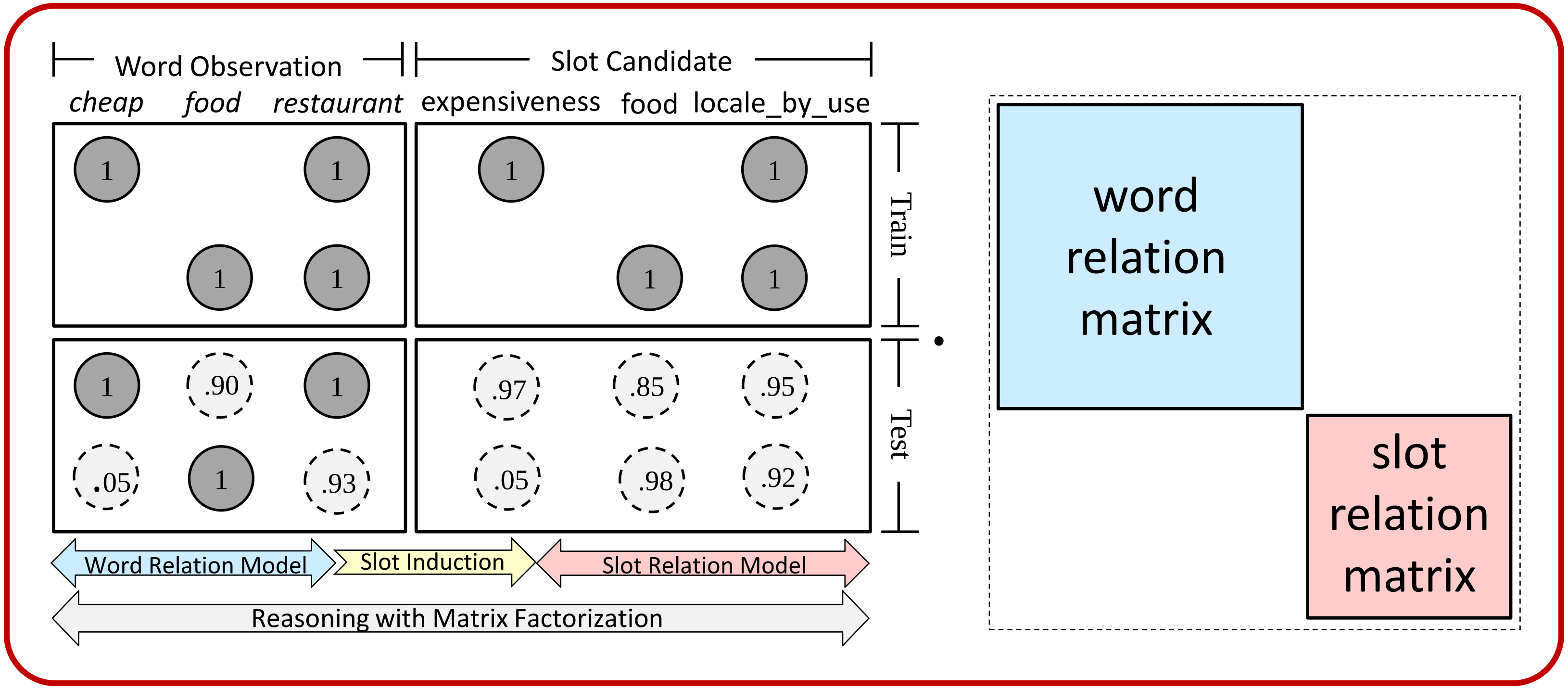
Test Utterance

show me a list of cheap restaurants

⋮



Matrix Factorization (MF)
Knowledge Graph Propagation Model



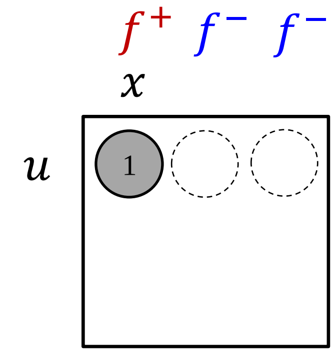
The MF method completes a partially-missing matrix based on the latent semantics by decomposing it into product of two matrices.

Bayesian Personalized Ranking for MF

Model implicit feedback

- not treat unobserved facts as negative samples (true or false)
- give observed facts higher scores than unobserved facts

$$\begin{aligned} f^+ &= \langle u, x^+ \rangle \\ f^- &= \langle u, x^- \rangle \end{aligned} \Rightarrow p(f^+) > p(f^-)$$



Objective:

$$\sum_{f^+ \in \mathcal{O}} \sum_{f^- \notin \mathcal{O}} \ln \sigma(\theta_{f^+} - \theta_{f^-})$$

The objective is to learn a set of well-ranked semantic slots per utterance.

Experiments of Semantic Decoding

Experiment 1: Quality of Semantics Estimation

Dataset: Cambridge University SLU Corpus

Metric: Mean Average Precision (MAP) of all estimated slot probabilities for each utterance

Modeling Implicit Semantics

Approach		ASR	Manual
Baseline: Logistic Regression		34.0	38.8
Random		22.5	25.1
Majority Class		32.9	38.4
MF Approach	Feature Model	**37.6**	**45.3**
	Feature Model + Knowledge Graph Propagation	**43.5** **(+27.9%)**	**53.4** **(+37.6%)**

The MF approach effectively models hidden semantics to improve SLU.

Adding a knowledge graph propagation model further improves the results.

Experiments of Semantic Decoding
Experiment 2: Effectiveness of Relations

Dataset: Cambridge University SLU Corpus

Metric: Mean Average Precision (MAP) of all estimated slot probabilities for each utterance

Approach		ASR	Manual
Feature Model		37.6	45.3
Feature + Knowledge Graph Propagation	Semantic Relation	41.4 (+10.1%)	51.6 (+13.9%)
	Dependent Relation	41.6 (+10.6%)	49.0 (+8.2%)
	Both	**43.5 (+15.7%)**	**53.4 (+17.9%)**

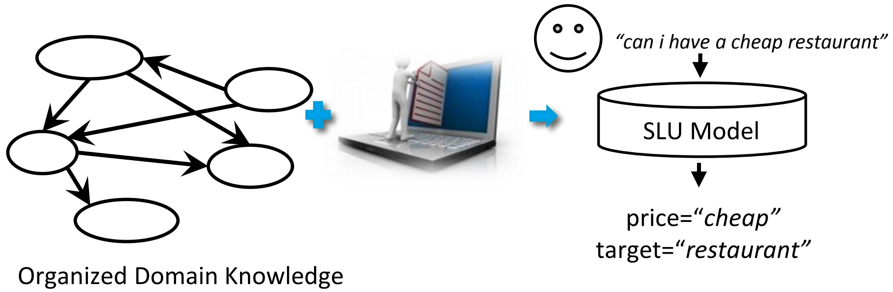
Both semantic and dependent relations are useful to infer hidden semantics.

Combining both types of relations further improves the performance.

Outline

- Introduction
- Ontology Induction [ASRU'13, SLT'14a]
- Structure Learning [NAACL-HLT'15]
- Semantic Decoding (submitted) }
- Conclusions

SLU Modeling



Outline

Introduction

Ontology Induction [ASRU'13, SLT'14a]

Structure Learning [NAACL-HLT'15]

Semantic Decoding (submitted)

Conclusions

Summary

Knowledge Acquisition

Ontology Induction → Semantic relations are useful

Structure Learning → Dependent relations are useful

SLU Modeling

Semantic Decoding

→ The MF approach builds an SLU model to decode semantics

Conclusions

The knowledge acquisition procedure enables systems to automatically learn open domain knowledge and produce domain-specific ontologies.

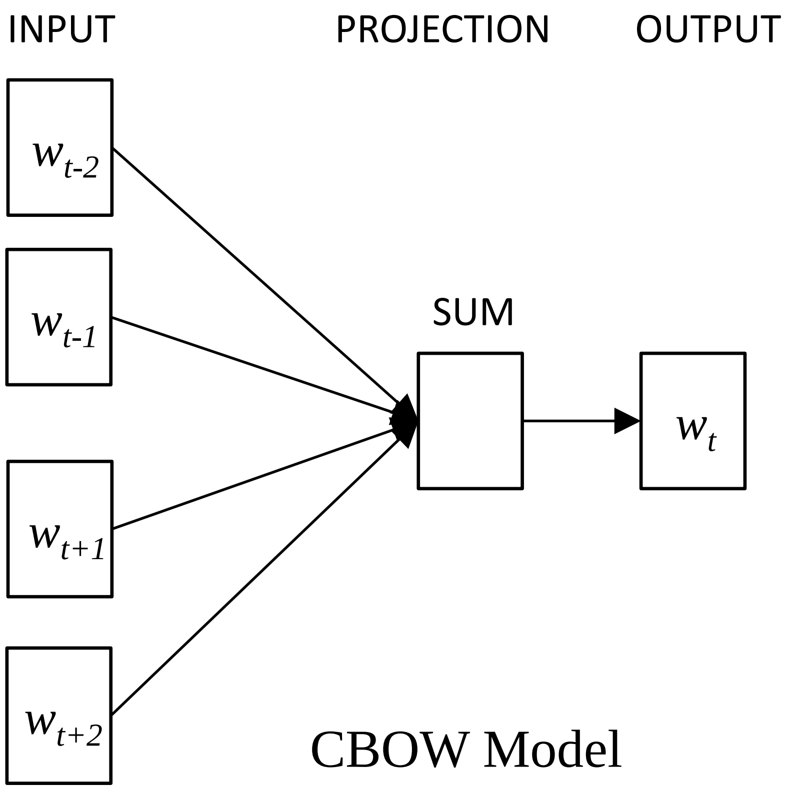
The MF technique for SLU modeling provides a principle model that is able to unify the automatically acquired knowledge, and then allows systems to consider implicit semantics for better understanding.

The work shows the feasibility and the potential of improving *generalization*, *maintenance*, *efficiency*, and *scalability* of SDSs.

Q & A

THANKS FOR YOUR ATTENTIONS!!

Word Embeddings



CBOW Model

Training Process

- Each word w is associated with a vector
- The contexts within the window size c are considered as the training data D
- Objective function:

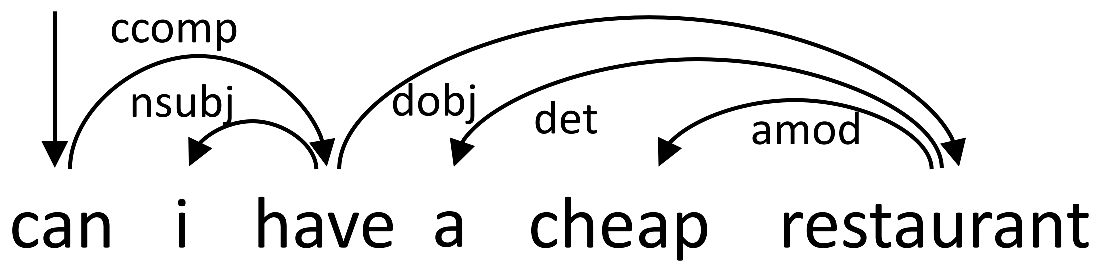
$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq i\leq c, i\neq 0}\log p(w_t \mid w_{t+i})$$

Mikolov et al., " Efficient Estimation of Word Representations in Vector Space," in *Proc. of ICLR*, 2013.
Mikolov et al., " Distributed Representations of Words and Phrases and their Compositionality," in *Proc. of NIPS*, 2013.
Mikolov et al., " Linguistic Regularities in Continuous Space Word Representations," in *Proc. of NAACL-HLT*, 2013.

[back]

Dependency-Based Embeddings

Word & Context Extraction



Word	Contexts
can	have/ccomp
i	have/nsub^{-1}
have	can/ccomp^{-1}, i/nsubj, restaurant/dobj
a	restaurant/det^{-1}
cheap	restaurant/amod^{-1}
restaurant	have/dobj^{-1}, a/det, cheap/amod

Levy and Goldberg, " Dependency-Based Word Embeddings," in *Proc. of ACL*, 2014.

Dependency-Based Embeddings

Training Process

- Each word w is associated with a vector v_w and each context c is represented as a vector v_c
- Learn vector representations for both words and contexts such that the dot product $v_w \cdot v_c$ associated with good word-context pairs belonging to the training data D is maximized
- Objective function:

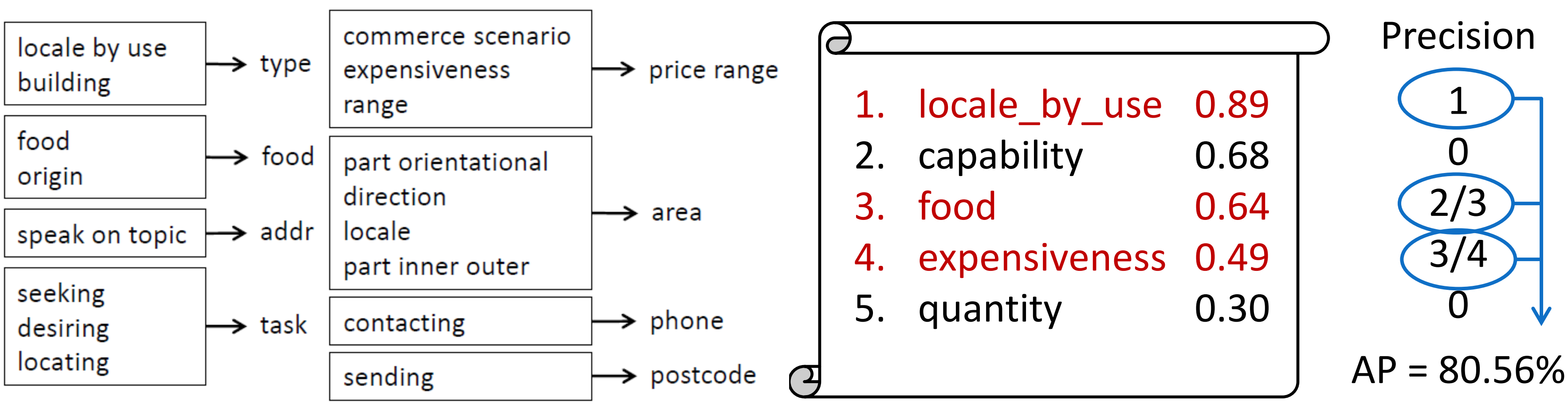
$$\arg\max_{v_w, v_c} \sum_{(w,c)\in D} \log \frac{1}{1+\exp(-v_c \cdot v_w)}$$

[back]

Levy and Goldberg, " Dependency-Based Word Embeddings," in *Proc. of ACL*, 2014.

Evaluation Metrics

- **Slot Induction Evaluation**: Average Precision (AP) and Area Under the Precision-Recall Curve (AUC) of the slot ranking model to measure the quality of induced slots via the mapping table



[back]

Slot Induction on ASR & Manual Results

The slot importance:

$$w(s) = (1 - \alpha) \log \underbrace{f(s)}_{\text{frequency}} + \alpha \cdot \log h(s)$$

ASR

1.	locale_by_use	0.88
2.	expensiveness	0.78
3.	food	0.64
4.	capability	0.59
5.	quantity	0.44

Manual

1.	locale_by_use	0.89
2.	capability	0.68
3.	food	0.64
4.	expensiveness	0.49
5.	quantity	0.30

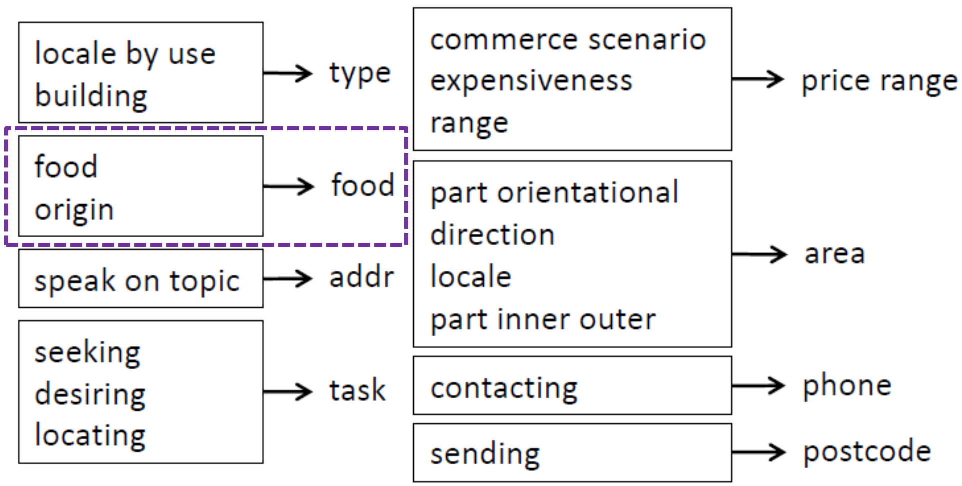
Users tend to speak important information more clearly, so misrecognition of less important slots may slightly benefit the slot induction performance.

[back]

Slot Mapping Table

Create the mapping if slot fillers of the induced slot are included by the reference slot

	induced slots		reference slot
	origin	**food**	**food**
u_1	asian	:	asian
u_2	:	beer	beer
:	:	:	:
u_k	japan	:	japan
:	:	:	:
u_n	:	noodle	noodle



[back]

Random Walk Algorithm

The converged algorithm satisfies

$$\begin{cases} r_s^* = (1-\alpha) r_s^{(0)} + \alpha L_{ss} L_{sw} r_w^* \\ r_w^* = (1-\alpha) r_w^{(0)} + \alpha L_{ww} L_{ws} r_s^* \end{cases}$$

$$\begin{aligned} r_s^* &= (1-\alpha) r_s^{(0)} + \alpha L_{ss} L_{sw} \left((1-\alpha) r_w^{(0)} + \alpha L_{ww} L_{ws} r_s^* \right) \\ &= (1-\alpha) r_s^{(0)} + \alpha (1-\alpha) L_{ss} L_{sw} r_w^{(0)} + \alpha^2 L_{ss} L_{sw} L_{ww} L_{ws} r_s^* \\ &= \left((1-\alpha) r_s^{(0)} e^T + \alpha (1-\alpha) L_{ss} L_{sw} r_w^{(0)} e^T + \alpha^2 L_{ss} L_{sw} L_{ww} L_{ws} \right) r_s^* \\ &= M r_s^* \end{aligned}$$

The derived closed form solution is the dominant eigenvector of M

[back]

SEMAFOR Performance

The SEMAFOR evaluation

Table 5
Frame identification results on both the SemEval 2007 data set and the FrameNet 1.5 release. Precision, recall, and F_1 were evaluated under exact and partial frame matching; see Section 3.3. **Bold** indicates best results on the SemEval 2007 data, which are also statistically significant with respect to the baseline ($p < 0.05$).

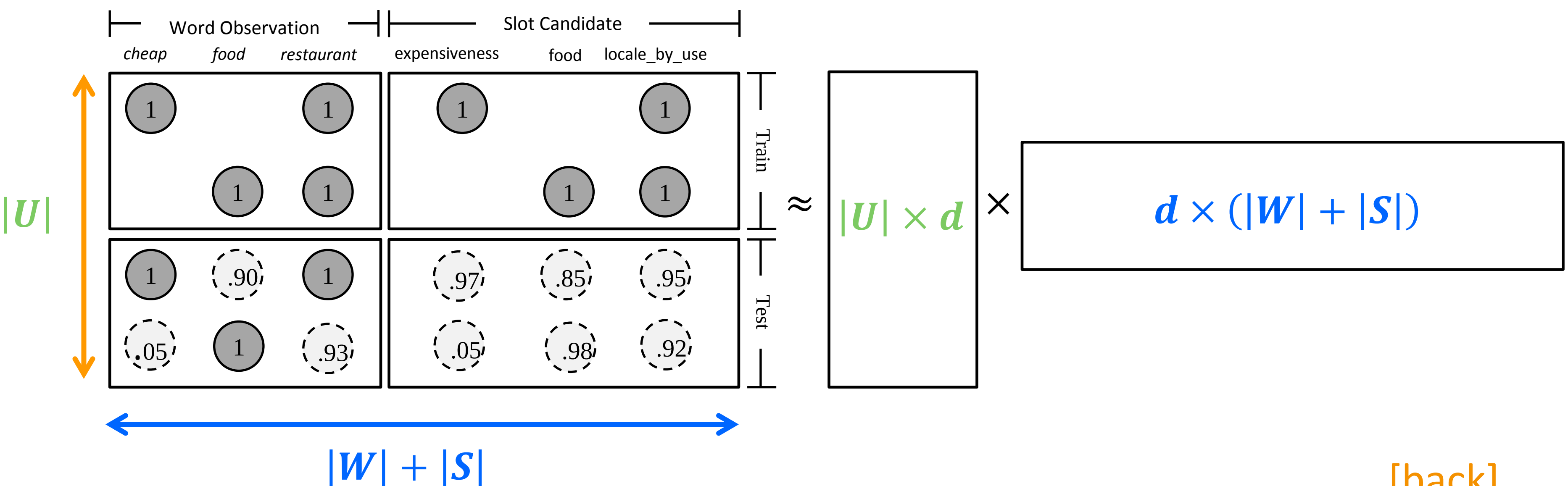
FRAME IDENTIFICATION (§5.2)		exact matching			partial matching		
		P	*R*	F_1	*P*	*R*	F_1
SemEval 2007 Data	gold targets	60.21	60.21	60.21	74.21	74.21	74.21
	automatic targets (§4)	**69.75**	**54.91**	**61.44**	**77.51**	**61.03**	**68.29**
	J&N'07 targets	65.34	49.91	56.59	74.30	56.74	64.34
	Baseline: J&N'07	*66.22*	*50.57*	*57.34*	*73.86*	*56.41*	*63.97*
FrameNet 1.5 Release	gold targets	82.97	82.97	82.97	90.51	90.51	90.51
	– unsupported features	80.30	80.30	80.30	88.91	88.91	88.91
	& – latent variable	75.54	75.54	75.54	85.92	85.92	85.92

[back]

Matrix Factorization

The decomposed matrices represent latent semantics for utterances and words/slots respectively

The product of two matrices fills the probability of hidden semantics



[back]

Cambridge University SLU Corpus

hi i'd like a restaurant in the cheap price range in the centre part of town	type=restaurant, pricerange=cheap, area=centre
um i'd like chinese food please	food=chinese
how much is the main cost	pricerange
okay and uh what's the address	addr
great uh and if i wanted to uh go to an italian restaurant instead	food=italian, type=restaurant
italian please	food=italian
what's the address	addr
i would like a cheap chinese restaurant	pricerange=cheap, food=chinese, type=restaurant
something in the riverside	area=centre

[back]