Lecture 16

Natural Language Programming with Virtual Assistants

I. Motivation
II. Two challenges
   • Capability + Training data acquisition
III. Key concepts
   • Virtual assistant programming language (DSL)
   • Grammar-based training data synthesis
IV. Privacy
Virtual Assistants

• Voice interface to our web accounts and IoTs
• Sees all account data & intermediates access
• Potential threat to our privacy & open competition
Virtual Assistant Oligopoly

- Alexa: 70% of the 76M installed base of owners in the US
- 100,000 3rd-party skills
- 60,000 compatible IoT devices
- 10,000 employees

A proprietary linguistic web is in the making!

Goal: Disrupt Surveillance Capitalism

1. Democratize voice assistant technology

2. **Thingpedia**: the best skill repository for all languages, open to all assistants.

3. **Genie**: A best and trustworthy virtual assistant
II. Two Challenges

(1) Beyond simple commands
Example: Asthma Patient

**Dr. Smith:**
- If Bob’s peak flow-meter drops below 180L/min, let me know.
- When the air quality index is above 500 and Bob is running, warn him.

**Bob:**
- Let my Dad know if I am at the hospital.

**Bob's environment:**
- When the air quality index is above 500 and Bob is running, warn him.

**Bob's devices:**
- When I use my inhaler, record my GPS location in logfile on Box.
More General than IFTTT

“*When I use my inhaler,*
*get my GPS location, if it is not home,*
*write it to logfile in Box.*”

- Event-driven program
- Multiple function calls
- Parameter passing
- Filters on values

Users have their own combination of interest.
Programming in natural language!
II. Two Challenges
(2) Acquisition of training data
Alexa’s Approach

Step 1: Alexa Meaning Representation Language (AMRL)

Step 2: Neural Network

Step 3: Interpreter

Step 4: Execute
Alexa: Syntax-Dependent Representation

Search for an upscale restaurant and then make a reservation for it

Reserve a high-end restaurant for me

Can you reserve a restaurant for me?
I want an upscale place.

找一家高档餐厅，然后帮我预约

我想预约一个高级餐厅

بگذارید ملاقات قرار بگیرم

یک رستوران خوب پیدا کنم و برای من قرار ملاقات بگذارید
Conventional Wisdom

• Natural language processing needs a neural network
• Neural network needs well annotated real users’ training data
  • Pre-requisite: Millions of real user
  • Coverage: Millions still don’t have enough coverage
  • Cost: 10,000 Alexa employees annotating real user data
  • Accuracy: 30% errors (Multi-Oz)
To address the challenges

We need a new methodology

1. Teach neural networks compositionality for natural language programming

2. Efficient acquisition of training data so we can scale to ask APIs and internet data
III. Key Concepts:
(1) Virtual Assistant Programming Language
Text

Search for an upscale restaurant and make a reservation for it

Meaning: Executable code

now => @com.yelp.Restaurant(),
price == enum(expensive)
=> @com.yelp.reserve(restaurant=id)

• Human Computer Interface is easier than NLP
• ThingTalk: Virtual Assistant (VA) programming language
  • Captures full capability of VA
  • A high-level language
  • Amenable to translation from natural language
  • An open, non-proprietary language fosters collaboration
• Can be extensible
Compositionality: Construct + API calls

```
When I use my inhaler, 
get my GPS location, if it is not home, 
write it to logfile in Box.
```

```
monitor @Inhaler-use(), 
=> @GPS(), location <> “home” 
=> @Box-write(file=“logfile”, data=location)
```
When the air quality index is above 500, and I am running, send me an SMS.
If I get taken to a hospital, let my dad know.
When I use my inhaler, get my location, save them to Dropbox.
If my heart rate is above 130, and I am not running, remind me to take a deep breath.
Thingpedia: Encyclopedia of Things

- Interoperability
  - API signatures + corresponding NL
  - Not just intents
- Open repository

<table>
<thead>
<tr>
<th>Twitter Event</th>
<th>Natural Language</th>
<th>API Signatures</th>
</tr>
</thead>
<tbody>
<tr>
<td>WHEN</td>
<td>@Stanford tweets</td>
<td>Monitor (@home_timeline(), …) author==“Stanford”)</td>
</tr>
<tr>
<td>GET</td>
<td>tweets matching “#Cardinal”</td>
<td>search(...), contains (hashtag, ...)</td>
</tr>
<tr>
<td>DO</td>
<td>tweet “Stanford won!”</td>
<td>post (status)</td>
</tr>
</tbody>
</table>
When my car is at home, and it is not plugged in, send me a reminder email.

Remind me if my car is not plugged in at home.
If I am not charging my car when it is home, let me know.
Remind me to charge my car whenever I’m home.
III. Key Concepts:
(2) Training data synthesis
Search for an upscale restaurant and then make a reservation for it.

Reserve a high-end restaurant for me.

Can you reserve a restaurant for me? I want an upscale place.

```
ThingTalk

now
=> @com.yelp.Restaurant(),
   price == enum(expensive)
=> @com.yelp.reserve
   (restaurant=id)
```
Could you please get me a restaurant that is upscale?

Reserve me a luxury restaurant

给我找一家高级餐厅并预约

Eʻimi i kahi hale ʻaina hulahula a laila
hana iā ia no ka mālama ʻana iā ia

高級レストランを検索してから予約する

Per favore riesci a trovarmi un ristorante?
Ho bisogno di qualcosa di lussoso.

Cerca un ristorante di lusso e dammi la prenotazione

Prenotami un ristorante da lusso
Semantic Parser Generation

### Natural Language API Signatures

<table>
<thead>
<tr>
<th>WHEN</th>
<th>@Stanford tweets</th>
<th>Monitor (@home_timeline(), ...)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>tweets matching “#Cardinal”</td>
<td>search(...), contains (hashtag, ...)</td>
</tr>
<tr>
<td>DO</td>
<td>tweet “Stanford won!”</td>
<td>post (status)</td>
</tr>
</tbody>
</table>

**Thingpedia**

**ThingTalk Grammar**

**NL Templates**

- When <> get <>
- Get <> when <>

**Synthesize programs**

**Paraphrase**

**Parameter & data augmentation**

**Validate & Test: Real User Input**

**Natural Language**

**LUInet**

**Code**
Neural Network

Natural Language

BERT

[CLS] find restaurants ...

Attention

LSTM

{SOS} Restaurant { ...}

ThingTalk

Embedding

Feed Forward

Final Vocabulary Distribution

γ Pointer Switch
Event-driven Commands

- 62% on real data
- Compositional sentences
- No training with real data
- Accuracy can be improved with more templates + paraphrases
- Extensions
  - Thingpedia + new constructs
## Applying Genie to Queries

<table>
<thead>
<tr>
<th>Queries</th>
<th>Alexa</th>
<th>Google</th>
<th>Siri</th>
<th>Genie</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show me restaurants rated at least 4 stars with at least 100 reviews</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Show restaurants in San Francisco rated higher than 4.5</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>What is highest rated Chinese restaurant in Hawaii?</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>How far is the closest 4 star and above restaurant?</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Find a W3C employee that went to Oxford</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Who worked for both Google and Amazon?</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Who graduated from Stanford and won a Nobel prize?</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Who worked for at least 3 companies?</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Show me hotels with checkout time later than 12PM</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Which hotel has a swimming pool in this area?</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>
Compound Restaurant Questions

415 Complex Questions

Alexa  Google  Siri  Genie
An Open Assistant

1000 Different IoT Devices with Home Assistant
1.8B Webpages schema.org data 1 domain a week
SmartNews Retail Music Banks

Thingpedia

Company's Website App Phone
amazon Google
Telcos Cable Co Autos Hotels
Messengers: Slack FB Whatsapp
Genie
IV. Privacy
Protect Privacy with an Open Federated Architecture

- **ThingTalk**: virtual-assistant programming language
- **LUInet**: open NLP neural network
  - training in the cloud (currently)
  - inference locally (in the future)
- **Genie**: Privacy-preserving assistant
  - Keeps users accounts & data local
  - Communicate/share with each other (like email)
  - Users share in natural language

"My dad can monitor my security camera only if I am not home."

A fully-functional research prototype is available as Almond for Android/web.
Sharing with Control in Natural Language

Requester: WHEN [FILTERS] → GET [FILTERS] → DO

FILTERS: =, <, >, <>, <=, >=, contains, starts with, ends with

Let Dr. Smith monitor my peak-flow-meter, if it drops below 180L/min.
Let my secretary, whenever I am out of town, read email messages whose subject is marked urgent.
Let CDC see any of my health records, provided my name is not included.
Let Websites know I am under 13 years old.

Protect privacy → Select data available for research
Remote Execution

(a) Request

“Ask @alice to notify me when her security camera detects motion.”

(b) Check

σ=＠dad, ε=SELF: monitor @security_camera.event(), has_motion=true ⇒ return

(c) Ask for permission

“@dad wants to get notified when any event is detected on your security camera and has motion is equal to true.”

(d) Respond

“Only if I’m not home.”

(e) Save

σ=＠dad, ε=SELF: monitor @security_camera.event(), has_motion=true && @phone.get_gps() {location≠home} ⇒ return

(f) Return detected events

“Notification from monitor security camera: motion detected ...”
Conformance of Access Controls

Natural Language
- Remote Program $p$
- Access control $c$

SMT Formula $L[p], L[c]$

Satisfiability Modulo Theories

Static conformance
- $p$ conforms to $c$ if $p \preceq c$
- $\equiv L[p] \models L[c]$
- $\equiv \neg \text{SAT}(L[p] \land \neg L[c])$

Dynamic conformance
- The program $p' = p \land c$
- is the least restrictive conforming program, provided $p' \neq \text{null}$.
Conclusions

- Highest programming language: natural language
- Target formal language: ThingTalk (a DSL)
  - Extensible: about 100 constructs?
  - Current research: schemas to dialogues
- Replace naive annotations with developer tools
  - Support millions of domain experts
  - Use grammar to generate training data to increase coverage
- Open-world collaboration to create an open linguistic web