INTRODUCTION AND MOTIVATION

- Almond is an open, crowd-sourced and programmable virtual assistant that we built
- General: built on an extensible library of devices "Thingpedia"
- Privacy aware: runs on private user phone
- Programmable: take commands in natural language executes them as trigger-action programs in a domain specific language
- Usable: fail-safe UI when natural language fails

DATA ACQUISITION

- **Training Data:**
  - Base data: a set of primitive sentences for each supported function in Thingpedia
  - Author data: 929 sentences of varying complexity that we wrote while testing Almond
  - IFITTT data: 5191 recipe descriptions for 64 recipes that we scraped from the ifttt.com website
- **Generated Data:** 1000 sentences that were generated mechanically by combining trigger and action
- **Paraphrasing Data:** 6333 sentences obtained by asking MTurk workers to paraphrase generated sentences
- **Test Data:**
  - Scenario & Composition data: data obtained from MTurk workers asked to come up with their own VA commands; 162 realistic sentences that capture linguistic variety and test unseen programs

THINGPEDIA

45 devices currently supported, over 200 functions

<table>
<thead>
<tr>
<th>Natural Language</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>WHEN</td>
<td></td>
</tr>
<tr>
<td>GET</td>
<td></td>
</tr>
<tr>
<td>DO</td>
<td></td>
</tr>
</tbody>
</table>

THINGTALK: A LANGUAGE FOR CONNECTING DEVICES

WHEN [FILTERS] → GET [FILTERS] → DO

When my alarm goes off, open my blinds
Get my tweets and save them to Dropbox
Every day at 9 am, get my tweets and send them to Dropbox

When I receive an email from my advisor then send the message on SMS

- play [presidential debate] from Youtube on my TV
- search video on youtube with query [presidential debate] then play url on tv with video url video url
- @twitter.search_video, query = [presidential debate], v0 = video_url
- @twitter.play_video, video_url = v0

MACHINE LEARNING ALGORITHMS

- **SEMPRE Algorithm**
  - Beam search based parser
  - Generate many programs and their canonical sentence
  - Log-linear model, with paraphrasing and lexical features
- **Seq2Seq with Attention**
  - 175-dim 1-layer LSTM

EVALUATION METRICS

- **Correct function**
  - High-level goals: Improve accuracy, coverage and extensibility
  - Accuracy: % sentences parsed to correct programs
  - We split our accuracy into correct function identification and the parameters
  - Parameters are hard to identify & literature fails to identify them at all
  - Coverage: % of programs that can be identified correctly by at least 1 sentence → Recall
  - Extensibility: % sentences/programs identified when tested on sentences from a new device or domain that had no manual training

ACCUACY

- **Coverage of Programs**
  - NN: covers more new programs with same training because it identifies compositions better
  - Linguistic variety in train set not enough to capture more realistic data

- **Composition**
  - 40
  - 51
  - 25
  - 49

EXTENSIBILITY

- **SEMPRE** is better than NN on new devices because it can identify incremental things with less training owing to good lexical identification of functions
- However, as compositions increase for new domain, NN’s compositional model beats SEMPRE

CONTRIBUTION OF TRAINING SETS

- We can increase our accuracy and recall as we add more linguistic variety and program diversity
- Paraphrasing has a mix of both while IFTTT and generated sets adds each element separately

CONCLUSIONS & FUTURE DIRECTION

- We believe that as we scale our VA, we need a more deep learned approach to sustain the performance
- The NN model has shown the potential to outperform the current SEMPRE-based system to identify a larger set of trigger-action programs but it needs better training data to learn a more robust linguistic model
- We hope to collect better training data by crowdsourcing the composition sentences and experiment more compositional NN models to learn new programs