Solving the Next Billion-People Privacy Problem

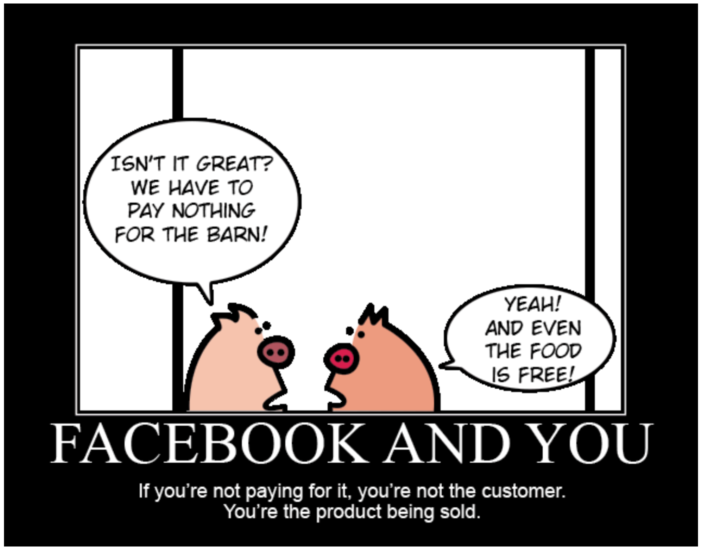
Monica Lam
Computer Science Department
Stanford University
lam@cs.stanford.edu

With Giovanni Campagna, Michael Fischer, Mehrad Moradshahi, Rakesh Ramesh, Richard Socher, Silei Xu, Richard Yang

Sponsors: AVG, Google, Hitachi, HTC, Hitachi, ING Direct, Nokia, Samsung, Sony Ericsson, UST Global

What is the Current
2 Billion-People Privacy Problem?

facebook


ISN'T IT GREAT? WE HAVE TO PAY NOTHING FOR THE BARN!
YEAH! AND EVEN THE FOOD IS FREE!
FACEBOOK AND YOU
If you're not paying for it, you're not the customer. You're the product being sold.

Geek and Poke: The Free Model

Terms of Service

"You own the content you create."

"You grant us a non-exclusive, transferable, sub-licensable, royalty-free, worldwide license to use any IP content that you post."

What is the Next
Billion-People Privacy Problem?

Virtual Assistants

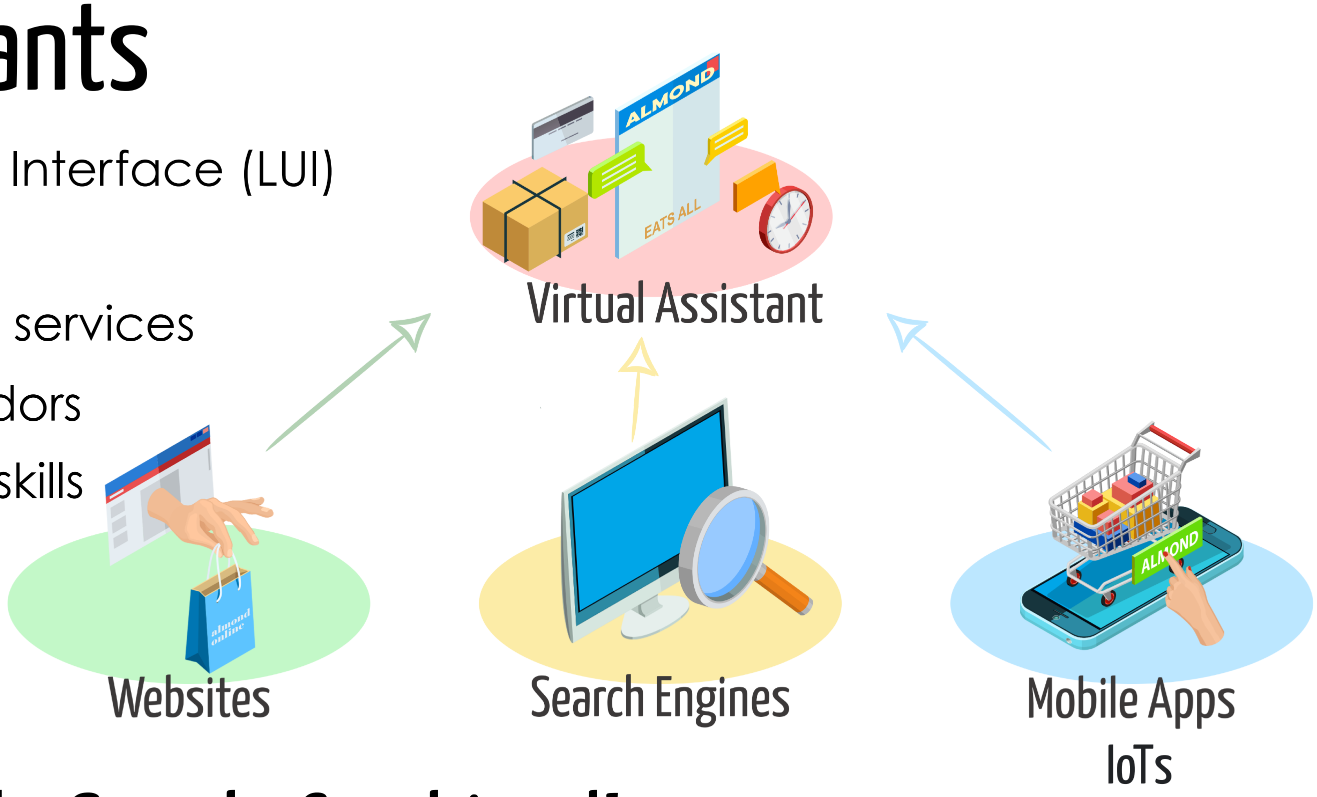
Personal, Linguistic User Interface (LUI)

Sees all personal info

Intermediates all digital services

Controls choice of vendors

Proprietary platform of skills



Virtual Assistant

Websites

Search Engines

Mobile Apps
IoTs

Amazon, Facebook, Google Combined!

What Can Academia Do? (1)

Tyranny of Convenience:

“Convenience has the ability to make other options unthinkable.”

Tim Wu, Columbia Law Prof., New York Times, Feb 2018

Make convenience a priority

Butler Lampson, SOSP keynote, 1999:

“Why didn’t we invent the web?”

What Can Academia Do? (2)

“Convenience and monopoly seem to be natural bedfellows”.

Tim Wu, Columbia Law Prof., New York Times, Feb 2018

Open, distributed architecture for privacy & open competition

Email: distributed consumer communication, 1972.

Turn Virtual Assistants into Our Friend

What we have done:

1. Develop open technology to
   make virtual assistants even more convenient!
2. Use virtual assistants to
   make sharing on distributed systems convenient!
3. Working prototype: Almond

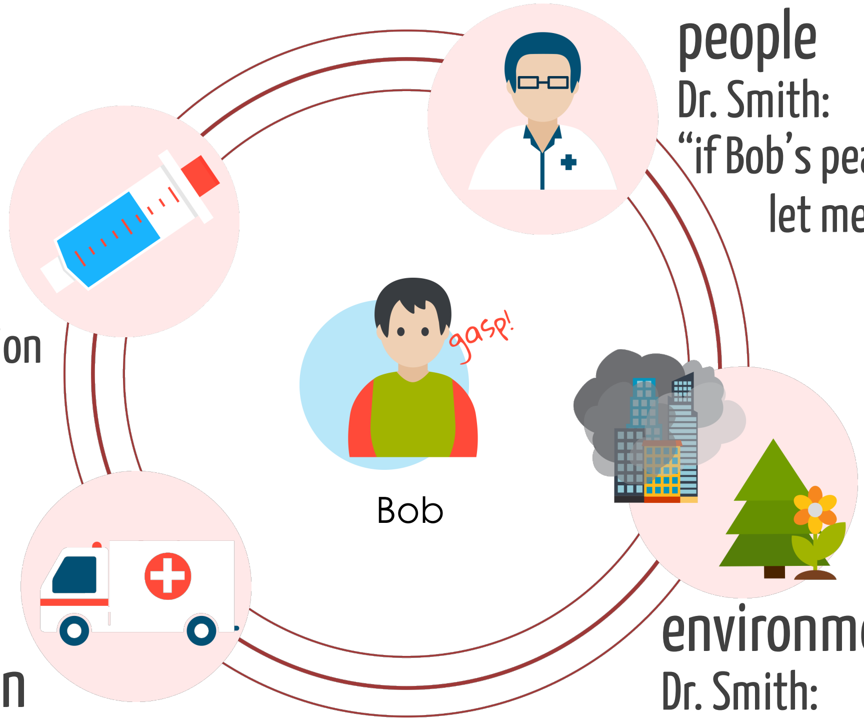
Next: Open Virtual Assistant Movement with academia & industry

Key Technology

Programming virtual assistants in natural language

Commercial Assistants	**Almond**
Only hardcoded skills	Can combine functions & add filters
Intent repository	API signatures repository
Dispatch model	IoT and services can inter-operate

Natural Language Programming: Asthma Example



people
Dr. Smith:
“if Bob’s peak flow-meter drops below 180L/min
let me know”

devices
“when I use my inhaler,
record my GPS location
in logfile on Box”

location
“Let my Dad know
if I am at the hospital”

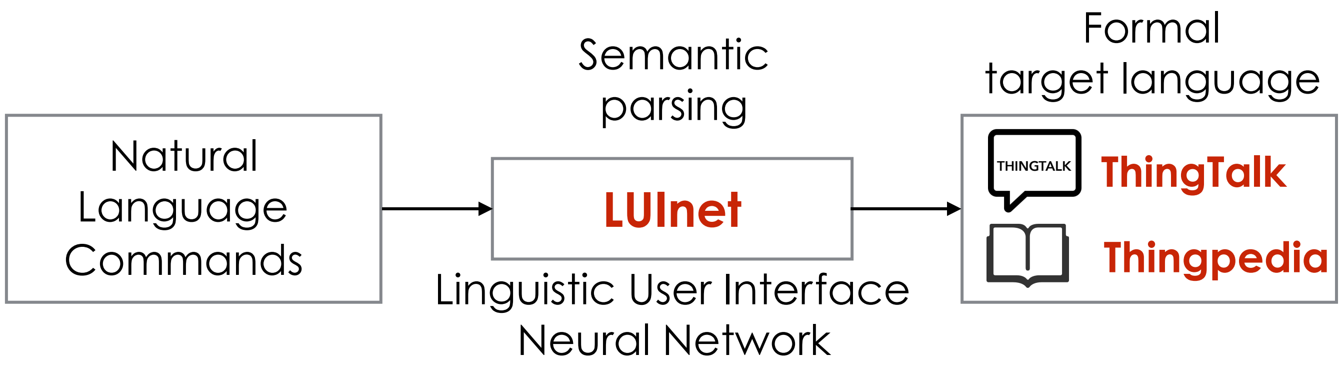
environment
Dr. Smith:
“when the air quality index is above 500 and
Bob is running, warn him”

Natural Language Programming

"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

Almond: 1st Programmable Virtual Assistant



"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)
```

Giovanni, Ramesh, Xu, Fischer, Lam, WWW 2017

Thingpedia: Encyclopedia of Things

> 60 devices / 200 functions

- Interoperability
  - API signatures + corresponding NL
  - Not just intent dispatches
- Open repository
  - Available to Alexa, Google Assistant, …

🐦	Natural Language	API Signatures
WHEN	@Stanford tweets	Monitor (@home_timeline(), …) author==“Stanford”)
GET	tweets matching “#Cardinal”	search(…), contains (hashtag, …)
DO	tweet “Stanford won!”	post (status)

ALMOND

Examples
Tweet the latest NASA Astronomy Picture of the Day
Auto reply to my emails
Set my phone to vibrate every day at 9 am
Turn on my TV if there is a person in the room
Translate Washington Post headlines to "Chinese"
Post my new Instagram pictures on Twitter
get a snapshot from my security camera every hour
Play some video from YouTube on my TV
Send me a daily cat picture

Almond Bike Market
WHEN: monitor second hand bike posts
WHEN: monitor bike posts of brand __
WHEN: monitor bikes for __
GET: search second hand bikes
GET: search bike posts of brand __
GET: search bikes for __
DO: post on almond bike market
DO: post a bike for __ dollars on almond bike market

Almond Dates
WHEN: monitor date posts on almond dates
WHEN: monitor date posts about __
GET: search partners on almond dates
GET: search __ partners
DO: post on almond dates
DO: post on almond dates to find partners for __

Bing Search
GET: search __ on bing
GET: search __ images on bing
GET: search images matching __ with size __ x __ on bing

Bluetooth Speaker
DO: set my speaker as default
DO: play music on my speaker
DO: increase volume on my speaker
DO: decrease volume on my speaker
DO: set volume on my speaker to __%

BodyTrace Scale
WHEN: my weight updates

Dropbox
GET: my dropbox quota
GET: file list in folder __ on dropbox
GET: file named __ on dropbox
DO: move file __ to __ in dropbox
DO: rename file __ in dropbox
DO: create a folder with name __ in my dropbox

Facebook
DO: post on facebook saying __
DO: post a picture on facebook
DO: post a picture on facebook with caption __

Giphy
GET: a ranDOm gif from giphy
GET: a gif with tag __ from giphy

Github
WHEN: a new issue opened in github repository __
WHEN: __ opens an issue on github
WHEN: user __ opens an issue in github repository __
WHEN: there is a new commit for github repository __
WHEN: user __ commits in github repository __
WHEN: a new milestone is created in github repository __
WHEN: user __ create a new milestone in github repository __
WHEN: there is a new comment in github repository __
WHEN: user __ comments on some issue in repository __
WHEN: there is a new comment on issue __ in repository __
DO: add email __ to my github account
DO: comment on issue __ in github repository __

Gmail
WHEN: receive an email on gmail
WHEN: receive an email from __ on gmail
WHEN: receive an email marked as important
WHEN: receive an email marked as important from __
WHEN: receive an email in category primary
WHEN: receive an email from __ in category primary
GET: the latest email
GET: the latest email with label __
GET: the latest email from __
GET: the latest email with subject __
DO: send an email to __ with subject __ with message __
DO: send a picture to __ with subject __

Google Drive
WHEN: a new file or folder is created on google drive
DO: create a new file with name __ on google drive

Holidays Calendar
WHEN: it's an holiday in the uk
WHEN: it's an holiday in the us
GET: the next uk holiday
GET: the next us holiday

iCalendar Events
WHEN: an event on my calendar begins
WHEN: an event on my calendar at location __ begins
WHEN: an event on my calendar organized by __ begins
GET: list my calendar events
GET: my calendar events organized by __
GET: my calendar events at __

Imgflip Meme Generator`
GET: all meme templates
GET: meme template named __
GET: generate meme on template __ with text __ at the top and text __ at the bottom

Instagram
WHEN: i upload a picture on instagram
WHEN: i upload a picture with filter __ on instagram
GET: my recent instagram pictures
GET: __ many recent instagram pictures
GET: my instagram pictures with filter __

Jawbone UP
WHEN: my steps on activity tracker updates
WHEN: i walked for __ steps
WHEN: i walked for __ distance
WHEN: my weight updates on my fitness tracker
WHEN: my weight is __ on my fitness tracker
WHEN: my bmi is __ on my fitness tracker
WHEN: my body fat is __ on my fitness tracker
WHEN: my heart rate updates
WHEN: my heart rate is __
WHEN: my sleep status updates on my sleep tracker
WHEN: i sleep for __ time

LG WebOS TV
DO: turn __ my lg tv
DO: raise the volume of my lg tv
DO: lower the volume of my lg tv
DO: set the volume of my lg tv to __
DO: mute my lg tv
DO: unmute my lg tv
DO: play link __ on my lg tv

LinkedIn
GET: my linkedin profile
DO: post __ on linkedin

Miscellaneous Interfaces
WHEN: it's __ o'clock every day
GET: current time
GET: current date
GET: give me a random number
GET: give me a random number between __ and __
DO: debug log __
DO: send me a message __

NASA Daily
WHEN: an asteroid passes close to earth
GET: nasa's astronomy picture of the day
GET: a picture from curiosity rover
GET: __ many pictures from curiosity rover
GET: a picture from curiosity rover taken on __

Nest
WHEN: the temperature on my thermostat updates
WHEN: the humidity on my thermostat updates
WHEN: there is a new event detected on my security camera
WHEN: my security camera detects something and has person is _
WHEN: my security camera detects something and has motion is _
WHEN: my security camera detects something and has sound is _
GET: the temperature on my thermostat
GET: the humidity on my thermostat
GET: the state of my hvac
GET: my security camera live feed
GET: me a snapshot of my security camera
DO: set temperature to __ on my thermostat
DO: set my temperature between __ and __ on my thermostat
DO: set my hvac to __
DO: turn __ my security camera

Omlet
WHEN: i receive a message on omlet
WHEN: i receive a message on omlet in feed __
WHEN: i receive a __ message on omlet
DO: send an omlet to __ saying __
DO: send a picture on omlet to __ with caption __

OneDrive
WHEN: a new file is created on onedrive
WHEN: a file is modified on onedrive
WHEN: file __ on onedrive is modified
DO: create a new file on onedrive named __ containing __
DO: delete __ from my onedrive
DO: rename __ to __ on my onedrive
DO: upload a picture to onedrive with name __

Parklon Iris Warm Water Mat
DO: turn __ my heatpad
DO: turn __ my parklon heatpad

PhD Comics
WHEN: there is a new post on phd comics

Philips Hue
DO: turn __ my lightbulb
DO: disco lights
DO: flash the lightbulb

Phone Companion
WHEN: my location changes
WHEN: i receive a sms
WHEN: i receive a sms from __
DO: show a popup with title __ and body __
DO: send an sms to __ saying __
DO: set my phone to __
DO: call number __
DO: call 911

Reddit Frontpage
WHEN: reddit front page updates
WHEN: a new post in category __ reaches reddit front page
WHEN: a new post from user __ reaches reddit front page

RSS Feed
WHEN: there is a new post on rss feed

Slack
WHEN: i receive a message on slack
WHEN: i receive a message from __ on slack
WHEN: i receive a message in channel __ on slack
DO: send a message on slack to __ saying __
DO: set the purpose for channel __ to __ on slack
DO: set the topic for channel __ to __ on slack
DO: set me as __ on slack
DO: send a picture on slack to __ saying __

SportRadar
WHEN: nba team __ plays
WHEN: nba team __ plays against __
WHEN: nba team __ plays and the game is __
WHEN: nba team __ __ a game
WHEN: eu soccer team __ plays
WHEN: eu soccer team __ plays against __
WHEN: eu soccer team __ plays and the game is __
WHEN: eu soccer team __ __ a game
WHEN: us soccer team __ plays
WHEN: us soccer team __ plays against __
WHEN: us soccer team __ plays and the game is __
WHEN: us soccer team __ __ a game
WHEN: monitor eu soccer games of tournament __
WHEN: monitor us soccer games of tournament __
WHEN: mlb team __ plays
WHEN: mlb team __ plays against __
WHEN: mlb team __ plays and the game is __
WHEN: mlb team __ __ a game .
WHEN: ncaa mens basketball team __ plays
WHEN: ncaa mens basketball team __ plays against __
WHEN: ncaa mens basketball team __ plays and the game is __
WHEN: ncaa mens basketball team __ __ a game .
WHEN: ncaafb team __ plays
WHEN: ncaafb team __ plays against __
WHEN: ncaafb team __ plays and the game is __
WHEN: ncaafb team __ __ a game

The Cat API
GET: a cat picture
GET: __ many cat pictures

The Wall Street Journal
WHEN: there is a new article in wsj opinions section
WHEN: there is a new article in wsj world news section
WHEN: there is a new article in wsj us business section
WHEN: there is a new article in wsj market news section
WHEN: there is a new article in wsj technology section
WHEN: there is a new article in wsj lifestyle section

The Washington Post
WHEN: there is a new article in washington post __ section
WHEN: there is a new blog post in washington post __ blog

Tumblr
WHEN: there is a new post in blog __ on tumblr
WHEN: there is a new picture uploaded in blog __ on tumblr
DO: post on tumblr with title __ and body __
DO: post __ on tumblr
DO: post a picture with caption __ on tumblr

Twitter
WHEN: someone i follow tweets
WHEN: user __ tweets
WHEN: someone replies to user __ on twitter
WHEN: i receive a direct message on twitter
WHEN: i receive a direct message from __ on twitter
WHEN: i tweet
WHEN: i reply to __ on twitter
GET: search for __ on twitter
GET: __ many recent tweets matching __
GET: recent tweets from __
GET: recent tweets from __ matching __
GET: recent tweets in reply to __
GET: recent tweets in reply to __ matching __
GET: search for tweets with hashtag __ on twitter
GET: __ many recent tweets with hashtag __
GET: tweets from __ with hashtag __
GET: tweets with hashtag __ in reply to __
DO: tweet __
DO: send a dm on twitter to __ saying __
DO: tweet a picture with caption __
DO: follow user __ on twitter
DO: unfollow user __ on twitter

Uber
GET: time estimate for uber
GET: give me a price estimate for uber from __ to __

Weather
WHEN: it's __ at location __
WHEN: monitor weather at __
GET: sunrise and sunset for location __
GET: sunrise and sunset for location __ on date __
GET: moon phase for location __
GET: moon phase for location __ on date __
GET: the weather in __

XKCD
WHEN: a new xkcd is out
WHEN: a new xkcd is out in the what-if section
GET: the latest xkcd
GET: the xkcd number __
GET: a random xkcd

Yahoo Finance
WHEN: the stock price of __ changes
WHEN: stock dividends for __ changes

Yandex Translate
GET: translate __ to __ with yandex
GET: translate __ from __ to __ with yandex
GET: translate __
GET: translate something to __
GET: translate with yandex from __ to __
GET: detect the language of __

Youtube
WHEN: there is a new video from youtube channels i follow
WHEN: there is a new video from youtube channel __
GET: list channels in category __ on youtube
GET: list channels i am subscribed to on youtube
GET: search __ channels on youtube
GET: search a __ video on youtube
GET: search a video from __ matching __ on youtube
GET: search __ many videos matching __ on youtube

LAM

STANFORD

THINGTALK

ThingTalk Compound Statement

WHEN [**FILTERS**] → **GET** [**FILTERS**] → **DO**

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

When I use my inhaler, get my location, save them to Dropbox

If I get taken to a hospital, let my dad know.

When the air quality index is above 500, and I am running, send me an SMS.

When the Bitcoin price reaches $10,000,
search for a "bitcoin" picture, and tweet it with caption "I'm rich!"

Real Natural Language Input

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 plug in my car whenever I'm home.

Technical Challenges

- **How to design ThingTalk + Thingpedia?**
  Usefulness. Synthesizability.
- **How to create an accurate LUInet model?**
  No real-life data. Compositional for scaling.
- **How to handle inaccuracy of LUInet?**
  GUI. Personal training.
- **How to teach users the scope of the formal language?**
  Popularity, auto-completion.

LUInet: Preliminary Results

Model:

Sequence-to-sequence neural network with attention.

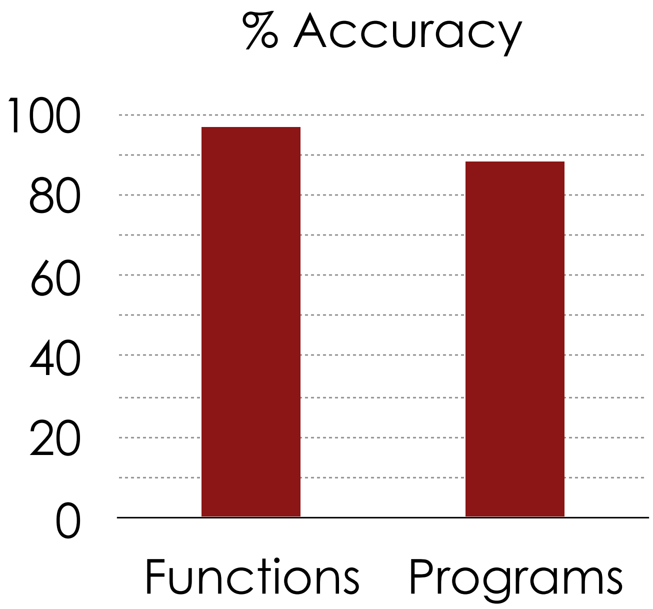
Bottom-up grammar productions.

Parameters are quoted:

‘Play the song “Born Free” on Spotify’

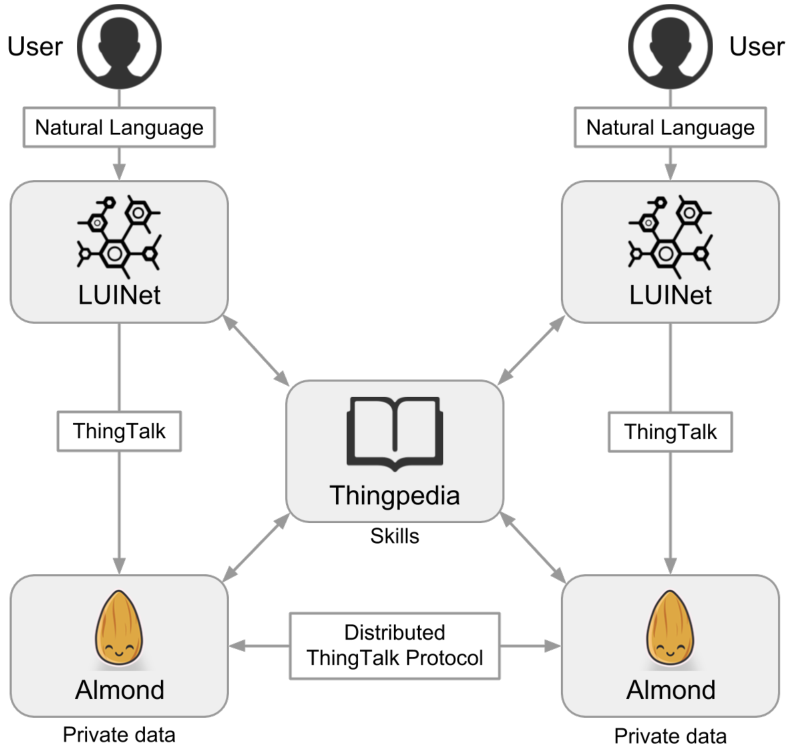
Dataset: 3410 programs

24566 sentences



Convenient Sharing with Privacy

Almond: 1st Distributed Virtual Assistant



Almond

Open source

Privacy: runs on our devices

Android app & web prototype

Distributed ThingTalk Protocol (DTP)

Interoperable assistants

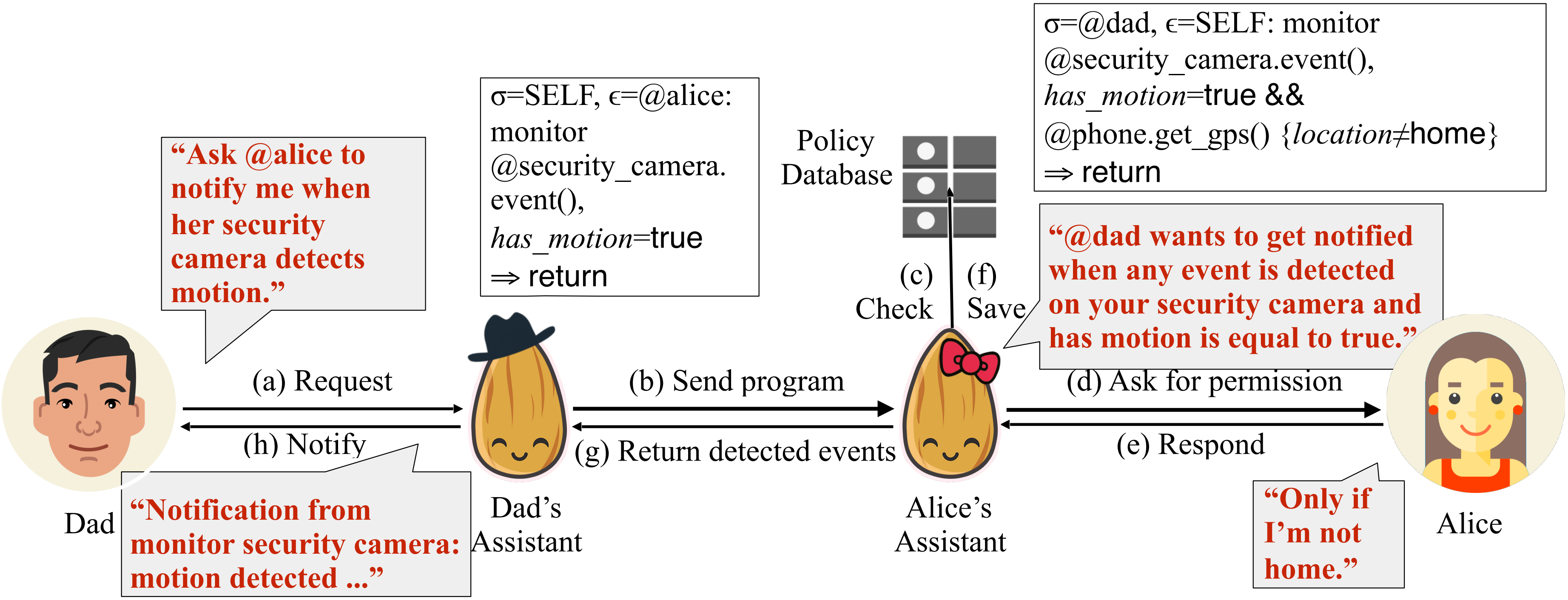
Fine-grain sharing with NL

Giovanni, Xu, Ramesh, Fischer, Lam, Ubicomp 2018

Virtual Assistant: Programmable Sharing Agent

Convenience	LUI (Linguistic User Interface)
Generality	Requests are ThingTalk programs
Control	Fine-grain access control in natural language. TACL formal language: a superset of ThingTalk
Privacy	Owner’s assistant executes requested ThingTalk, returns only allowed results
Security	Execute precisely approved program
Conformance correctness	TACL —> SMT (Satisfiability Modulo Theories)

Example



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

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

Policy Database

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

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

(c) Check

(f) Save

(a) Request

(h) Notify

(b) Send program

(g) Return detected events

(d) Ask for permission

(e) Respond

Dad

Dad’s Assistant

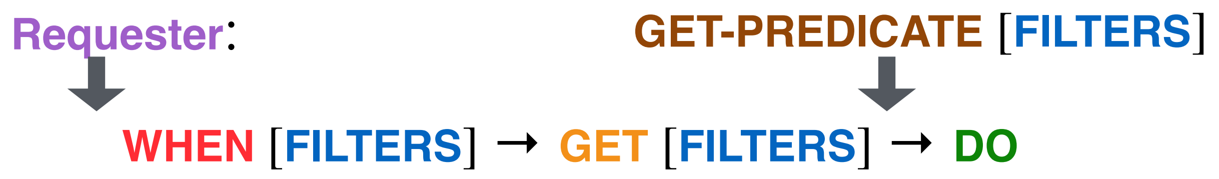
Alice’s Assistant

Alice

“Notification from monitor security camera: motion detected ...”

“Only if I’m not home.”

TACL: ThingTalk Access Control Language



Requester:

GET-PREDICATE [**FILTERS**]

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 father, monitor my security camera for motion, only if I'm not home.

Let my accountant get my monthly statements

Let my daughter, from 6-8pm, watch NetFlix

Let my boyfriend, get pictures from my dropbox, taken on Feb 14, and post them on Facebook

TACL Parsing Accuracy

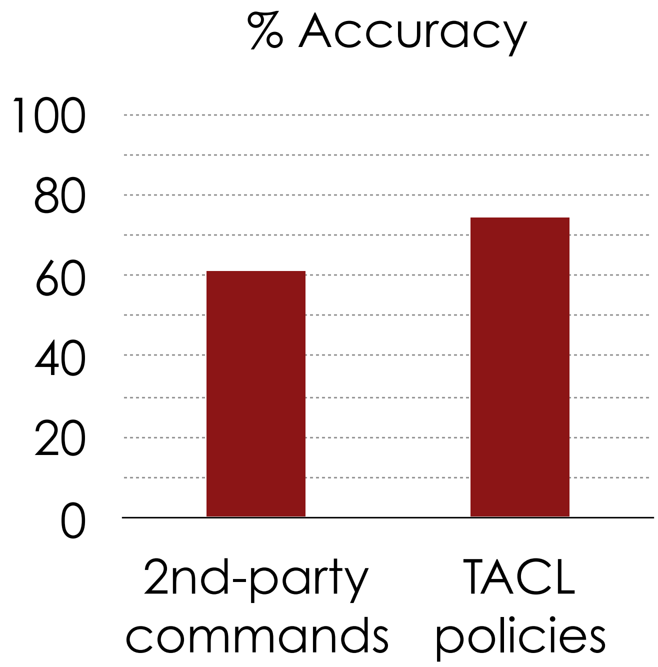
Model:

Sequence-to-sequence neural network with attention.

Bottom-up grammar productions.

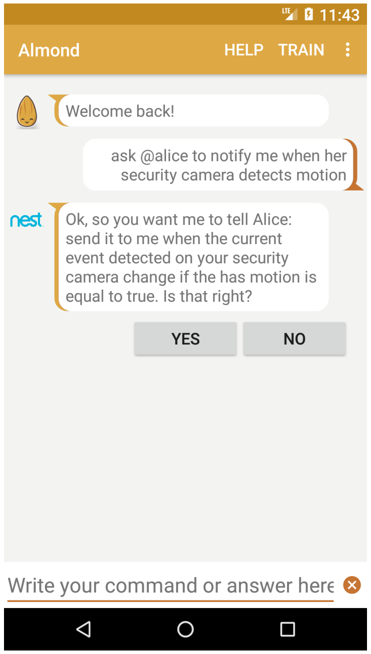
Parameters are quoted

+Dataset:	3577	2nd-party commands
	4285	TACL policies



Automatically Generated GUI

Dad
wants access

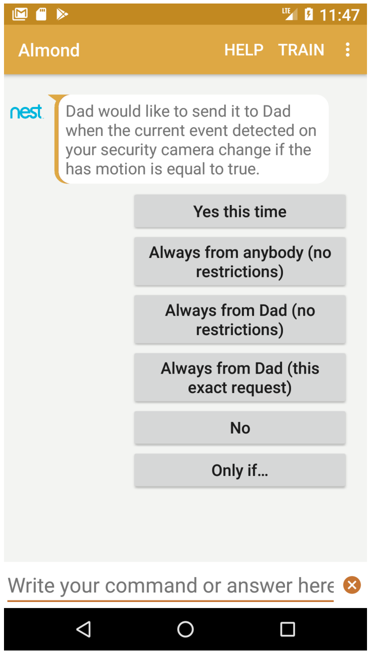


Welcome back!

ask @alice to notify me when her security camera detects motion

nest: Ok, so you want me to tell Alice: send it to me when the current event detected on your security camera change if the has motion is equal to true. Is that right?

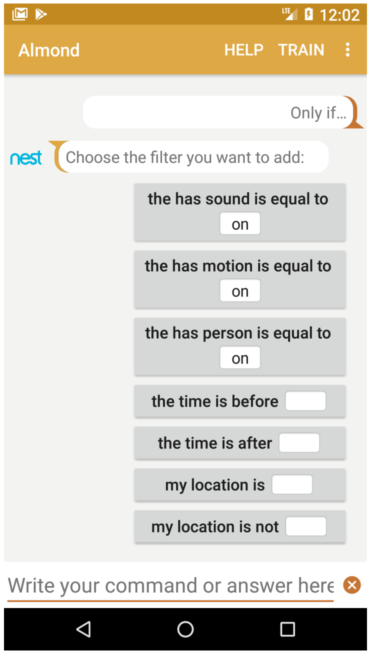
→ Ann
approves?



nest: Dad would like to send it to Dad when the current event detected on your security camera change if the has motion is equal to true.

Yes this time
Always from anybody (no restrictions)
Always from Dad (no restrictions)
Always from Dad (this exact request)
No
Only if...
Write your command or answer here

Ann
restricts access

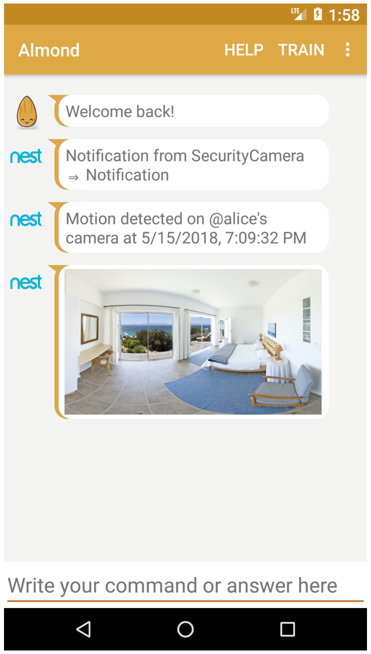


Only if...

nest: Choose the filter you want to add:

- the has sound is equal to on
- the has motion is equal to on
- the has person is equal to on
- the time is before
- the time is after
- my location is
- my location is not

Dad
gets result

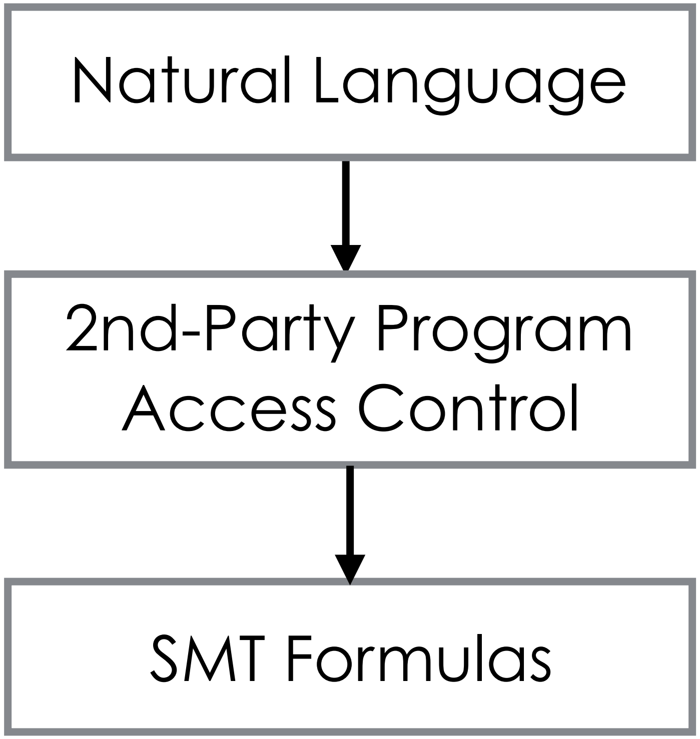


Welcome back!

nest: Notification from SecurityCamera ⇒ Notification

nest: Motion detected on @alice's camera at 5/15/2018, 7:09:32 PM

Formal Access Control



Natural Language

2nd-Party Program Access Control

SMT Formulas

SMT (Satisfiability Modulo Theories)

Bob can post cat pictures from Instagram to Twitter

σ = @bob :
    monitor @instagram.get_pictures(),
        contains(*hashtags*, #cat)
$\Rightarrow$ @twitter.post_picture(
        url = *instagram.url*,
        caption = "cat")

TACL (c)

σ = @bob $\wedge$

$(Y_{1,\text{url}}, Y_{1,\text{hashtags}}) = F_{\text{instagram.get\_pictures}}(r_1) \wedge$

$\text{mkHashtag(“cat”)} \in Y_{1,\text{hashtags}}$

$\wedge\ X_{\text{D,url}} = Y_{1,\text{url}}$

$\wedge\ X_{\text{D,caption}} = \text{“cat”}$

SMT Formula ($L[c]$)

Conformance of Access Controls

Conformance

p conforms to c if $p \preceq c$

$$\equiv L[p] \vDash L[c]$$

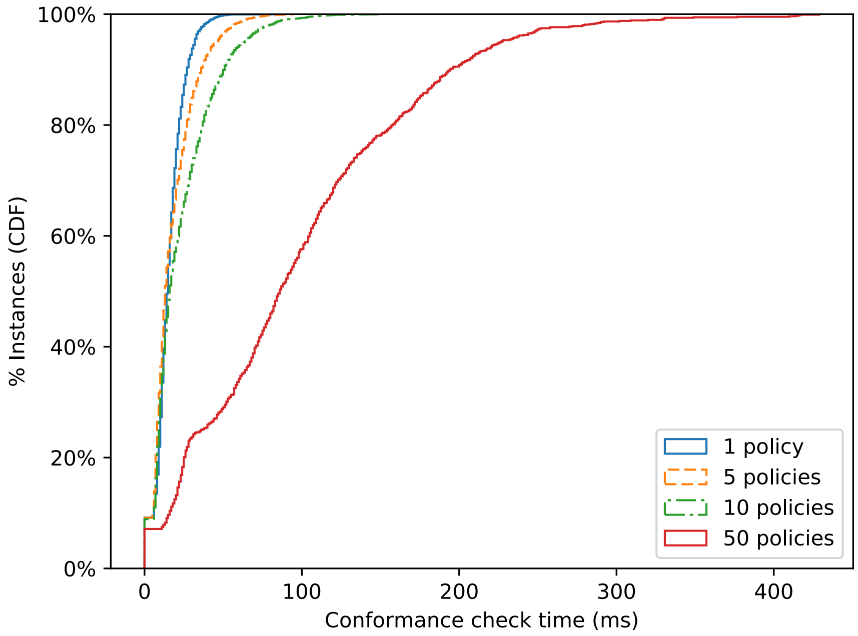
$$\equiv \neg\, \mathrm{SAT}(L[p] \wedge \neg L[c])$$

Synthesis of a conforming program

The program $p' = p \wedge c$

is the least restrictive conforming program, provided $p' \neq$ null.

SMT is Fast Enough

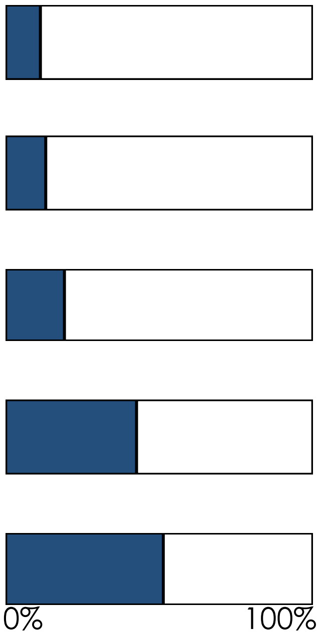


- CVC4 SMT checker, v1.5
- 2.5 GHz Intel Xeon CPU, 80 GB RAM
- 50 policies allowing same functions run in 0.4 seconds

Needs and Acceptance?

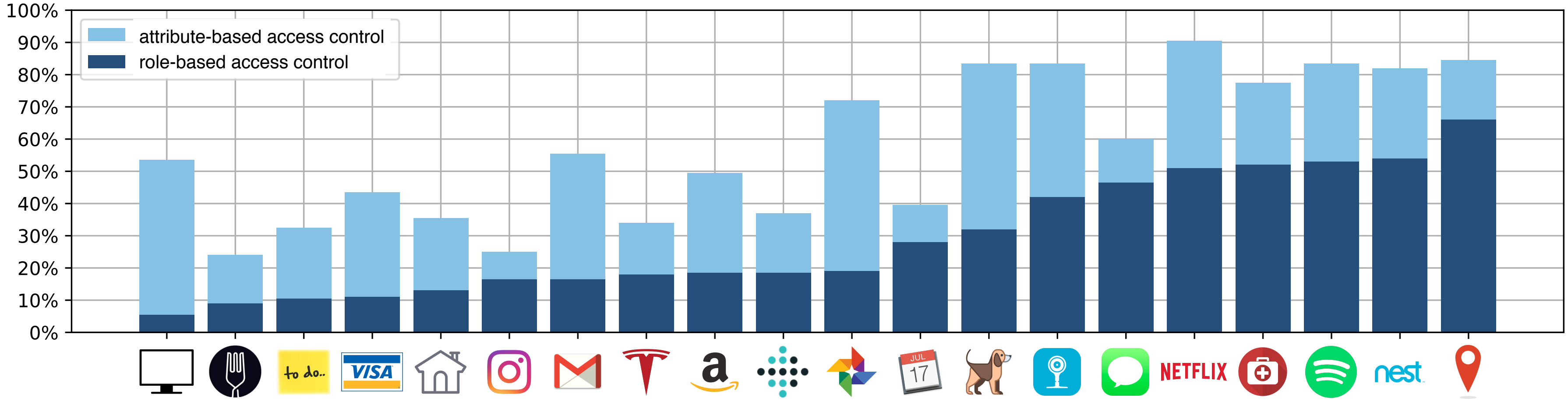
Do Consumers Need Access Control?

Role-Based Permission ■	🟦 Attribute-Based Permission
Teenage daughter to use credit card	With a $20 budget limit For restaurants only
Amazon courier to unlock door	If the package is over $1000 If your security camera is on
Friends to access cloud drive	Photos with their faces in them Photos in a specific folder
Parent/kid to see security cameras	If you are not at home Cameras facing the front yard/garage
10-year-old kid to use Netflix	Between 7 PM to 9 PM Free G or PG rated movies



% People comfortable in giving permission (200 person survey)

More Examples



Willingness to share doubles with attribute-based access control

Expressiveness of TACL

Solicit use cases by showing AMT workers 3 examples,
without describing ThingTalk or TACL

Enforceable:

Mom: "You need to follow this guy on Twitter, give me your Twitter account".

Me: "OK, add him but don't follow any other twitter user".

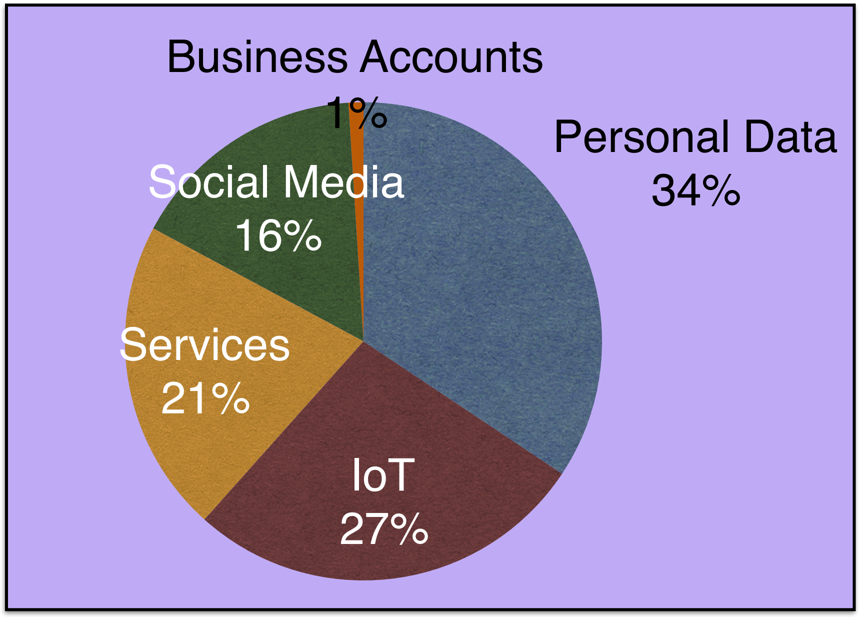
Unenforceable:

Friend: "Can I use your library card?"

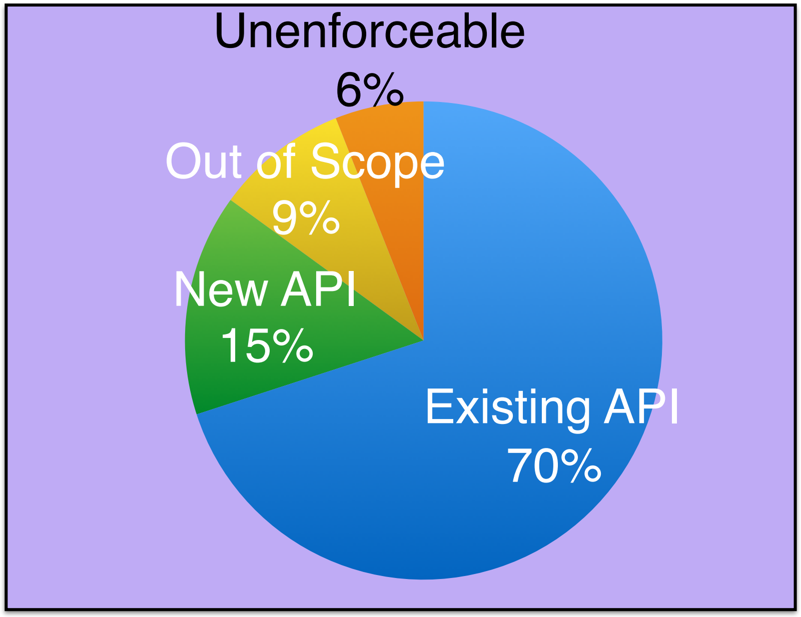
Me: "OK, only if you return the book on time".

TACL is Expressive Across Diverse Uses

60 workers; 220 suggestions; 85 unique assets

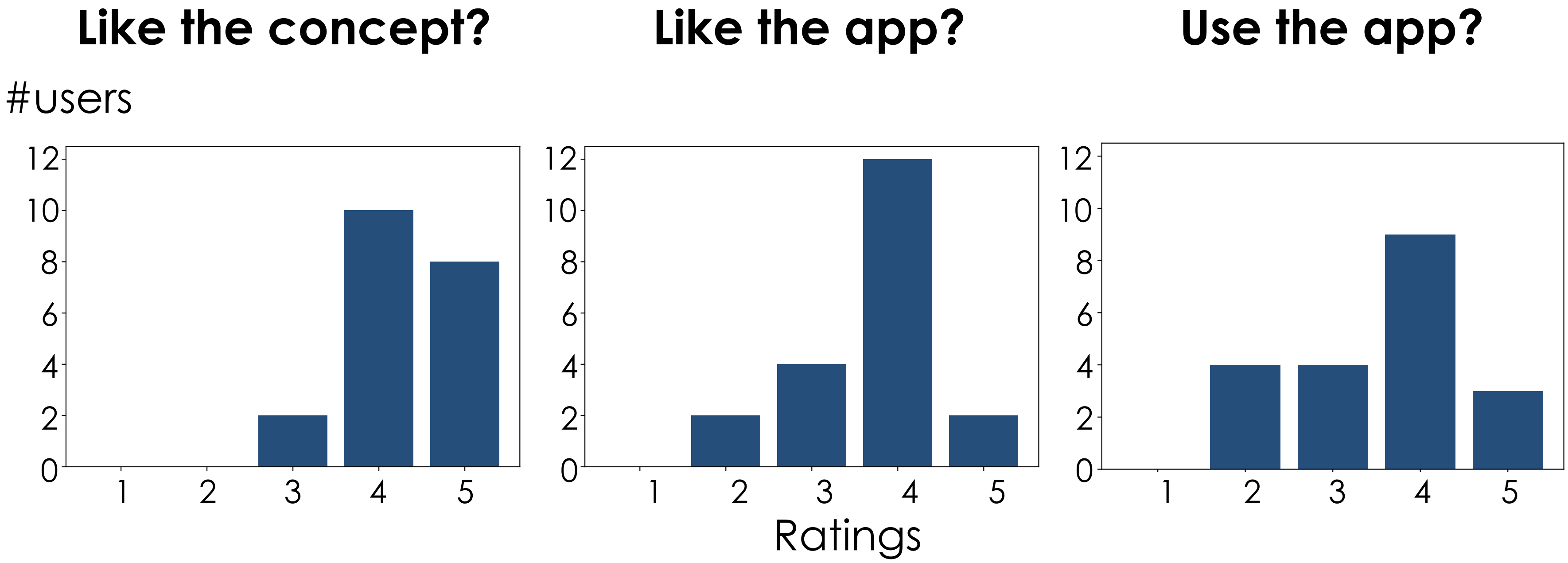


Diverse use cases



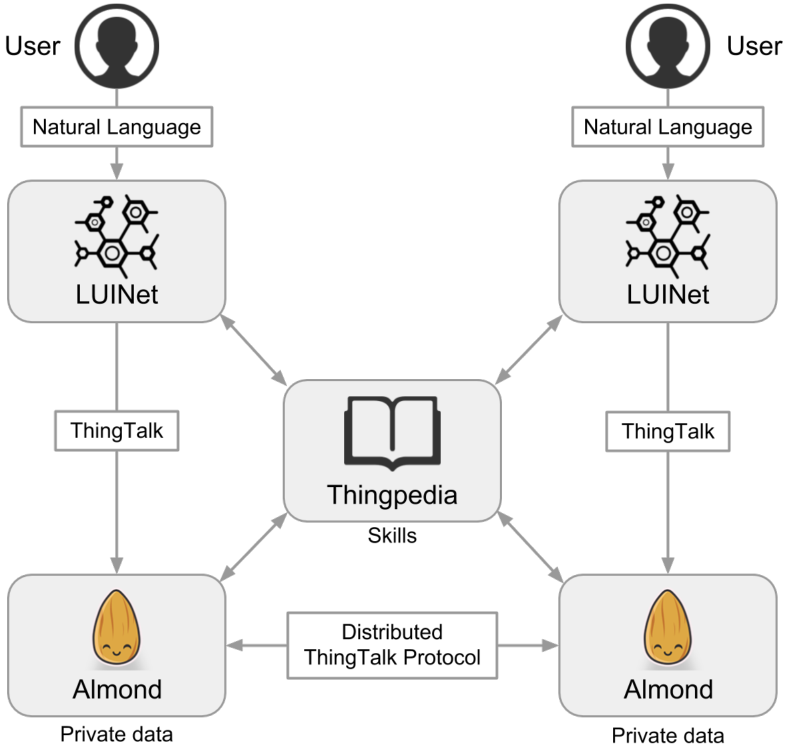
85% in the scope of TACL

User Study: “Sharing Without Passwords”



Summary

- **Tyranny of convenience**
- **Open competition: pre-requisite to privacy**
- **Natural language programming** (ThingTalk, TACL)
- **Communicating virtual assistants:** convenient fine-grain sharing
- **Open software movement:** Thingpedia, LUInet, Almond, DTP


User
User
Natural Language
Natural Language
LUINet
LUINet
ThingTalk
ThingTalk
Thingpedia
Skills
Almond
Private data
Distributed ThingTalk Protocol
Almond
Private data