Service Matching in Ad Hoc Agent Systems

Cactus Plenaire Vergadering/Mini Conference
W.Pasman
3 December 2003
Overview

1 State of Art in Service Matching
   Smarthome HAL: commercial system
   Chu-Carroll: Call Routing
   CSLR Phoenix parser
   Context handling in Story Understanding

2 Cactus Testbed Architecture
   Organizing agents
   Service Matching
Smarthome HAL

Voice/Script based system, X10 powerline communication
Control via RC/mic/(mobile)phone
X10 Device ID

Device Wizard

HAL will automatically control the device(s) that have a matching X-10 code. Select the House Code and Unit Code for the device.

House Code: A01
Unit Code:

Device Wizard

HAL will allow you to turn appliances on and off, dim lights, open and close doors, and more!

Select the type of action this device performs:

- On/Off (e.g., lights and appliances)
- Dimmable (lights)
- Open/Unlock (powered doors/locks)

Special controls
Give the device a name
You use this name to address the device
Special agents (weather, sports, stocks,..)

Focusing command eg ‘Select TV Listings’

After that you can ask ‘what’s on TV tonight’ etc.

Agents have config screen
‘Complex’ scripts with command and precondition
Call Routing

Setting: large company (e.g., bank) wants to forward incoming calls automatically to the right department

(Chu-Caroll '99)
1. Map incoming call to a vector
2. Determine closest existing service vector
3. Make disambiguation query if more than 1 vector close

Implementation: Corpus + statistics
Corpus: 4500 calls to 23 departments of bank
Found 757 n-grams used in first utterance.
Create ~20dim vector spaces spanning the services
Sing Val Decomposition of ngram+service matrix needed.
Phoenix Parser

Frame-based Parsing:

Frames with Slots

Skip un-recognised inputs
Early approach (Constantinides):
All parses are scored
Maximize number of covered words
Minimize number of fragments
Maximize number of filled slots in a frame

-> Not suited for many different services
Later improvements (Rudnicky):

Slots are pre-sorted to their ‘power to bring solution’

Task tree to guide the dialog,

For instance trip=fly+hotel+car. top node (a) tries to gather all data, (e) is currently active getting a plane and (i) is trying to find out the departure date.
Story Understanding

Several approaches.
Marker passing approach inspiring for Cactus system

Example (Norvig). User talks about fishing and island. Body of water is common parent --> notion of location, surrounded by and body of water are ‘inferred’.
Various types of links are used to derive the right relations. Eg constraints, subclass-of, slot-of, view (alternative interpretation), quantifier.

Larger nets allow more extensive use of context.

However large nets cause huge number of such shared parents

One approach: only certain type of link combinations are relevant.

Later research indicates that statistics (by adding probabilities to edges) can greatly help searching for relevant relations
CACTUS. Organizing ad-hoc agents

Every agent knows its place: task- location- and user- relations, and the relevance of the relation
Some details

Agents change the context by changing the relations in the agent world.
Every agent also knows how the area in the physical world it ‘represents’
Service Matching

**Service Matching**: search agent matching user’s need

- Some agents in the agent world understand NL
- Context-dependent search using agent’s context info
- Using Natural Language request from user
- Searching agent space for agent understanding request
- Gradual extension of search space
- User’s request is also used to start up the service (no repeat of the request necessary)
Conclusion

We are working on and improving
• distributed natural language understanding
• highly adaptable systems with large number of services
Literature


