Interpreting instructions in a pedestrian routing domain

Jana Götze[1], Johan Boye[1], Morgan Fredriksson[2], Jürgen Königsmann[2]
[1]KTH, School of Computer Science and Communication, 100 44 Stockholm, Sweden
[2]Liquid Media, Hammarby allé 34, 120 61 Stockholm, Sweden
{jagoetze, jboye}@kth.se {morgan, jurgen}@liquid.se

1 Introduction

Simulated users are important vehicles for testing, development and evaluation of dialogue systems.

We describe an implementation of simulated users that can interact with a dialogue system for pedestrian routing and exploration via written natural language. To emulate real user behaviour, such simulated pedestrians need to have a representation of a user's goals, the past dialogue history, the geographic context, as well as capabilities for generating realistic movement patterns, and for contextually interpretating route instructions.

The dialogue system, described in (Boye et al. 2012) and henceforth called R, uses data from the OpenStreetMap (OSM) geographic database (Haklay, 2008) to construct a route from the user's starting position to his goal, and then give instructions as the user is moving.

A key problem for the simulated user is to interpret such instructions and to resolve the references to objects in the city. Such references form the link between the algebraic and geometric model of the domain, and the communication with the user. For the simulated user it is crucial to correctly interpret instructions like "Turn left onto King's Street.", or questions like "Can you see the statue?" to be able to follow them. In order to interpret questions like the latter, it is also important to have access to information on visibility. In our simulation, visibility is checked on the basis of the OSM database, by continuously calculating whether the line of sight between the user's position and the surrounding objects is intersected by another object such as a building.

2 The pedestrian routing domain

Routing systems have been around quite some time for car navigation, but the pedestrian routing problem is different and in many senses more difficult, as pedestrians have many more options to choose from. Pedestrian routing systems have recently been studied by several researchers (Bartie and Mackaness, 2006; Krug et al. 2003; Janarthanam et al. 2012).

R employs a dialogue strategy of first grounding landmarks with the user, and only then use them in routing instructions. We now want a simulated user that can hold up the user's end of the dialogue to generate dialogues like the following:

1. System:	There is a fountain about 35 metres from here. Can you see it?
2. User:	Yes.
3. System:	Good! Please walk to the left of the fountain.
4. User:	(walks)
5. System:	Please turn right and walk to the top of the stairs.
6. User:	I cannot see any stairs.

In order to generate behaviour that resembles that of a real pedestrian, our simulated user S has a representation of the direction S is currently heading, the desired direction, the set of landmarks currently visible, the landmarks that have been mentioned in previous utterances, and the places that have been visited on previous occasions. S also maintains a representation of the objects in the immediate vicinity in order to generate movement, and to understand relative references like "left" and "right", and a representation of landmarks in its field of vision, but a complete knowledge of the entire city is neither necessary nor desired. The restricted geographic knowledge of the simulated user mimics that of a real pedestrian.

3 Interpretation of utterances

A semantic parser translates natural-language utterances into context-independent expressions in a flat meaning representation language, which is then further processed to resolve references and generate context-dependent interpretations. On the basis of these, goals can be added to the queue of actions for the simulated user to do next.

Here, we consider instructions and propositional questions that require geographical context to find an appropriate referent, as well as utterances that additionally require dialogue context.

For instance, the instruction "Turn left at the junction towards Starbucks on East Crosscauseway", is represented by:

```
dialogAct(inform, X),
X : turn(left, A, B, C),
isA(A, junction),
isA(B, cafe),
isNamed(B, starbucks),
isA(C, street),
isNamed(C, 'eastcrosscauseway')
```

In this expression, the variable X is a handle that acts as pointer to the succeeding expression turn(left, A, B, C). The use of handles is inspired by minimal recursion semantics (Copestake et al. 2005). The variables A, B and C are implicitly lambda-bound, and the purpose of the spatial reference resolution mechanism is to find the identifiers of the nodes that the speaker referred to.

The key semantic predicate for instructions is turn(Dir, TurningPoint, AimPoint, Street)

The values of the arguments are constrained by the instruction. The utterance above constrains all four, whereas "Turn left" only constrains the first, and "Go towards Starbucks" only the third.

In order to find concrete nodes to fill in the TurningPoint argument, the set of nodes visible from the user's position, and the set of nodes visible from the next goal node is calculated, and a node matching the description is sought among these nodes. The landmarks that AimPoint and Street refer to are not required to be in view, so the whole set of nearby nodes is searched.

The resolved utterance then becomes:

```
dialogAct(inform, X),
X : turn(left, 21135018, B, 23614881),
isA(21135018, junction),
isA(2156953057, cafe),
isNamed(2156953057, 'starbucks'),
isA(23614881, street),
isNamed(23614881, 'eastcrosscauseway')
```

where the lambda-bound variables of the unresolved expression have been substituted with identifiers of nodes and ways. These in turn will be added to the queue of short-term goals. In this example, the user is asked to first go to the junction, and then towards the cafe, i.e. first the junction with ID 21135018, denoting the TurningPoint will be added, then the AimPoint.

4 Behaviour generation

The simulated user S generates movement and dialogue behaviour. Dialogue acts that can be expressed are requests for directions ("Directions to Camera Obscura"), requests for instructions ("Where should I go now?"), answers to specific questions ("Yes, I can see Starbucks"), acknowledgements ("Okay"), reports of miscommunication ("I didn't understand that"), reports of success ("Thanks, I can see Camera Obscura"), and a few others.

A dialogue always begins with the simulation stating the long-term goal, e.g. "Directions to Camera Obscura". It then starts walking in a random direction awaiting the first instruction which will lead to one or several short-term goals being put on the goal queue if the instruction is interpreted successfully. If reference resolution does not result in any matching object, a miscommunication report will be generated (e.g. "Go to Starbucks" – "I don't know where Starbucks is."). If S receives no instructions, it will try to guess an appropriate next short-term goal and put it on the queue on its own initiative. Most often, S will continue walking in roughly the same direction as before, but with a small probability it will deviate from its current course and randomly select a new direction.

In addition, the simulated user has a scalar representation of how assertive it is that the current direction is correct. This assertiveness is increased if a given instruction can be interpreted sensibly, e.g. when the instruction is "Turn left", and it is indeed possible to turn left at the next short-term goal. If it is not possible to turn left, assertiveness will be decreased. The assertiveness is also slowly decreased as time elapses without it having received a route instruction, and even more so if S needs to change direction on its own initiative. If the assertiveness value falls below a certain threshold, this information can be used to generate a request for help, e.g. "Where should I go now?".

5 Concluding remarks

We have described an implementation of simulated users in a pedestrian routing domain that can interpret route instructions in their spatial context and dialogue context. Ongoing work includes methods for also modifying the dialogue behaviour using past interactions with real users as well as testing how a simulated user interprets instructions that real users give to describe routes while moving along them, e.g. using the corpus described in (Albore et al. 2013).

References

Albore, A., Boye, J., Fredriksson, M., Götze, J., Gustafson, J. and Königsmann, J. (2013) Final Pedestrian Behaviour Component. Deliverable 2.3.2, SpaceBook project, http://www.spacebook-project.eu/pubs/D2.3.2.pdf.

Boye, J., Fredriksson, M., Götze, J., Gustafson, J. and Königsmann, J. (2012) Walk this way: Spatial grounding for city exploration. *Proc. IWSDS workshop*, Paris, France. Extended version to appear in *Towards a Natural Interaction with Robots, Knowbots and Smartphones*, Springer-Verlag, 2013.

Bartie, P. and Mackaness, W (2006). Development of a speech-based augmented reality system to support exploration of cityscape. *Transactions in GIS*, 10(1):6386.

Copestake, A., Flickinger, D., Pollard, C., and Sag, I. A. (2005). Minimal recursion semantics: An introduction. Research on Language and Computation, 3(2-3), 281-332.

Haklay, M. (2008) OpenStreetMap: User-generated street maps. *Pervasive computing IEEE*, vol. 7, issue 4, pp. 12–18.

Janarthanam, S., Lemon, O. Liu, X., Bartie, P., Mackaness, W., Dalmas T. and Götze, J. (2012), Integrating location, visibility, and Question-Answering in a spoken dialogue system for pedestrian city exploration. *Proc. SeineDIAL, 16th Workshop on the Semantics and Pragmatics of Dialogue*, Paris, France.

Krug, K., Mountain, D. and Phan, D. (2003) Webpark: Location-based services for mobile users in protected areas. *GeoInformatics*, pp. 26–29.