Presenting a system of human-machine interaction for performing map tasks.

Gabriele Pallotti*, Francesca Frontini, Fabio Affè**, Monica Monachini**, Stefania Ferrari***
* Università de Modena e Reggio Emilia, **ILC CNR Pisa
name.surname@unimore.it, **name.surname@ilc.cnr.it

Abstract
A system for human machine interaction is presented, that offers second language learners of Italian the possibility of assessing their competence by performing a map task, namely by guiding the a virtual follower through a map with written instructions in natural language. The underlying natural language processing algorithm is described, and the map authoring infrastructure is presented.

Keywords: Language learning, human machine interaction, map tasks

1. Introduction

The present paper presents an infrastructure for the interactive performance of map tasks for assessing competence in Italian as a second language.
In a map task (Figure 1) a person (*giver*) is asked to describe in a natural language a path drawn onto a map to another person (*follower*) provided with the same version of the map where no path is drawn. The second person will try to reproduce the same path onto his/her map.
Map tasks are a valid and reliable data elicitation method used in several types of linguistic research. They belong to a larger family of "information-gap tasks" (Mackey & Gass 2006), which have been shown to lead speakers to produce relatively long and comparable speech samples in a natural and non-contrived way. The giver's turns are basically monological but there is room for clarification questions and comprehension checks by the follower, which makes these tasks ideal for collecting both sustained monological speech and interactional exchanges. Interaction can be further increased by providing the two participants with two slightly different maps, leading to a higher number of negotiation sequences. Map tasks are thus interesting in that they show how "how speakers jointly construct talk around route directions using a map tas"(Filipi ed al, 2004), achieving an optimal balance between sustained and interactive speech.
Map tasks have been employed in several large-scale projects comparing speech by different participants from different areas and with different characteristics (e.g. HCRC Map Task Corpus, Anderson et al 1991; Interactive atlas of Catalan intonation; AVIP Project - Vocal Archives of Spoken Italian, n.d.). They are also widely used in second-language acquisition research for comparing performance in a variety of communicative tasks (e.g. Robinson 2001; Gass et al 2005; Gilabert et al 2009). Although not yet widely used in language testing and assessment, map tasks have been identified as a promising tool in the area of performance-based language assessment (Norris et al 1998; Bachman 2002).

2. Our idea

We describe here a first attempt at a computational version of the traditional map task test in which a computer performs the role of the "follower". The developed interface is meant to be used mainly for practicing and assessing Italian as a second language in the context of an online learning environment - the LIRA web platform, currently being developed by the universities of Perugia for foreigners, Verona, Bologna and Modena and Reggio Emilia.
This kind of task can be seen as a very simplified two dimensional version of ongoing works in artificial intelligence, enabling robots to understand natural language directions and move accordingly in space (see among others Kollar 2010).
Maps and tasks may be more or less complex, based on the user's linguistic competence. Complexity can be increased by adding more landmarks on the map, by making paths less rectilinear, by making more specific requests as to the trajectory to be completed.
In the play interface the user is provided a map on which a path is drawn, connecting two points. S/he must instruct the machine to follow it, by typing a series of directions in natural language. Instructions are followed by minimal backchannelling feedback (in the form of a confirmation message such as "ok") by the computer when they can be parsed without problems. When the computer cannot parse the input or cannot perform the action, it provides feedback in the form of clarification requests or comprehension checks (Gass 1997), such as for instance *Non riesco a capire* ("I cannot understand"), or *Non posso andarci* ("I can't go there").
The computer can interpret directions based on punctual landmarks, such as buildings and crossings. This will allow it to interpret instructions like *prendi via Mazzini e continua fino alla scuola* ("take the third street to your right and continue until the school"). The performance of the human giver is determined based on whether the target is reached and how many instructions were needed to get there.

3. Navigation module

The algorithm needs to read each input sentence, identify the destination point, verify if the destination point is reachable and, if so, add the destination point to a data structure containing the path. For instance if at a certain point of the path the instruction is *gira a destra e prosegui fino allo stadio* (turn right then go straight till the stadium), the computer needs to check - given its current position - whether there is a street on the right and if there is a stadium on that street.

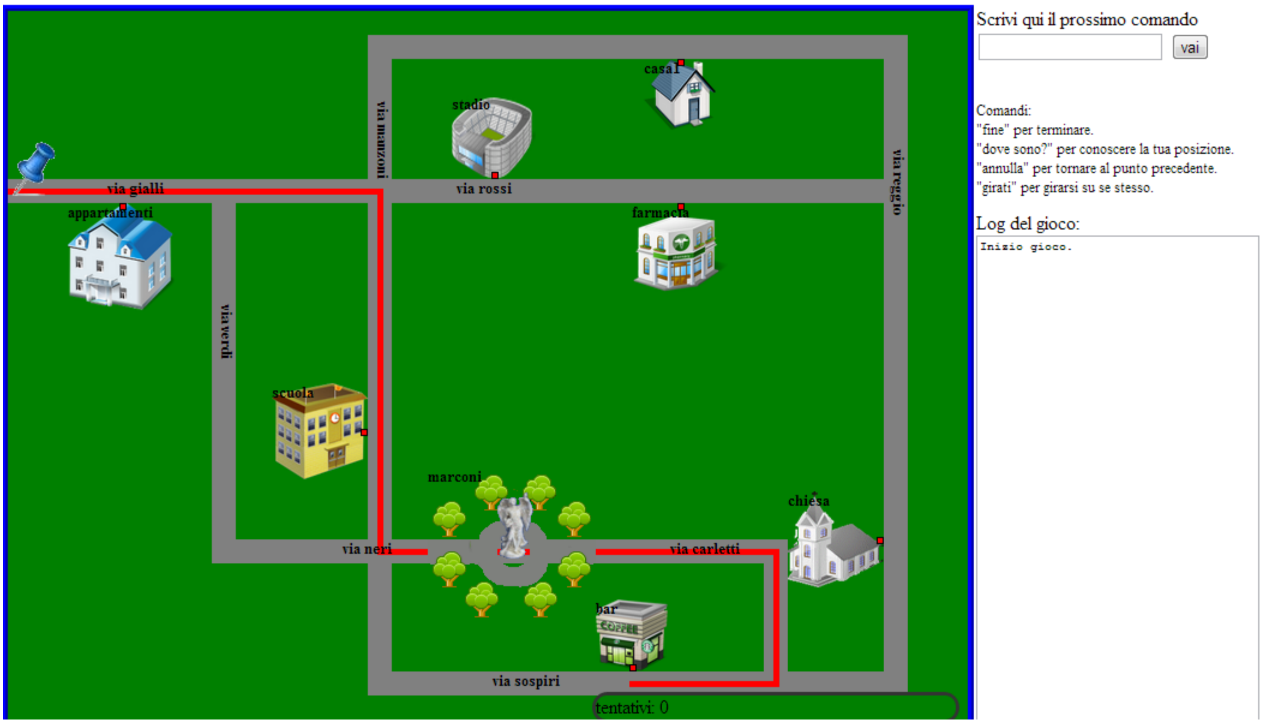

Figure 1: Results at different cutoffs for the U test, using a PSC based set of lemmas.

In the current version of the map every crossing and building has been identified as a possible target. For each instruction the system needs to identify the direction in which it is asked to move and the point it is asked to stop at. Before moving in the right direction the system must also check whether obstacles are present, since moving through buildings is not allowed. Instructions that "collide" with buildings are recorded as errors and do not produce any progression in the path. In order to add complexity to the task the algorithm could be later modified in order to enter and exit buildings and to navigate through maps of increased complexity that require diagonal movements.
A special problem that needs to be dealt with is perspective taking. In the present version the choice was for the internal perspective. The algorithm needs to interpret every instruction as if they were directions given to a human on the street. A small arrow is drawn on the map to clarify this perspective, and directions must be given with respect to that point of view. Thus movements on the x and y axis can be expressed by a "turn right/left" or "go straight" instruction, depending on the direction previously taken by the follower.
The navigation algorithm functions on the basis of a set of minimal commands that navigate the pointer on the map; when the sequence of valid commands does not allow for any valid movements on the present map, the system outputs a negative feedback such as *Non posso andarci* ("I cannot go there").

4. Natural Language Processing module

The navigation module requires an NLP module to translate natural language instructions into the basic commands. Understanding instructions in a natural language is a hard artificial intelligence task. Yet the kind of natural language used in map tasks is a simpler subset of what people normally produce in everyday conversations. Moreover, the written medium, as opposed to the spoken language normally used in traditional map task, should increase the level of standardization and predictability of the input.
The analysis of map task dialogues between human beings shows that both native and non-native speakers of Italian produce a relatively small set of patterns of this kind (examples are given in English for brevity):

- continue on this street
- turn right at crossing
- then hospital
- then shop
- go to university
- ...

The current version of the algorithm simply parses the input in search of instructions containing directions and landmarks. In the given examples one can easily translate the sentences into instructions such as:

- [go] straight
- [turn] [right] at the [first] [crossing]
- [continue] till [hospital]
- [continue] till [shop]
- [go] to [university]

- ....

Syntactic parsing was not considered therefore necessary, nor was proper PoS tagging. A simple stemming and a rule based algorithm can parse most sentences in real time. The algorithm recognizes a basic set of instructions such as the ones listed above by going through a decision tree:

1 identify the basic action to perform (continue/turn/reach)

2a if the action is a continue action, the end point must be identified by searching for a landmark. If no landmark (such as a building) is present, then the system moves onto the next crossing or to the end of the street

2b if the action is a turn action, the system must identify at which crossing to turn (check for counters) and in which direction to turn (right/left)

If the decision tree returns a negative result (as it happens when no basic action is identified, or when no target is present), the linguistic processor returns a linguistic feedback, *Non capisco* (I cannot understand you), without passing any instruction to the navigation module.
More complicate instructions could be of the form:
''Keep going until you reach the church, then continue and then turn right"
In this case the machine needs to parse the conjunctions ('then', 'and'...) in order to recognize that two instructions are given, with three directions and three target points. The present algorithm performs a split on a finite set of conjunctions (and/then/...) and then treats the result as a set of consecutive instructions.
At present the minimal instructions are treated using a bag of wordS?** approach, also in order to minimize the impact of learner's syntactic/word-order errors; thus the system:

- has a basic lexicon of go/continue-verbs, turn-verbs and possible buildings/landmarks, that was expanded using ItalWordnet (Roventini et al. 2003)
- can count till four
- recognizes other expressions such as *fino a* (until).

Special rules for crossing a square (attraversa la piazza), for making a u-turn (*girati*) and for exiting the game (*fine*) are implemented. Undoing the last command is also possible (*annulla*). Currently the system cannot deal with with mis-typing in the input. Word similarity algorithms such as Edit Distance could be put in place to try and recognize the misspelled word, yet this might slow down the interaction. Once a set of common errors is collected from a history of games, an efficient solution will be implemented. Another possible improvement could be the implementation of feedback by the machine ("did you mean ...?")., in order to make communication more interactive. Here again a set of possible corrections can be crowdsourced from players.
In order to collect data for research and for the improvement of the algorithm, a log is kept of each game session.
Finally, some shallow morpho-syntactic parsing, although not necessary in the identification of target points, may be used later for evaluating the linguistic proficiency of the giver. This analysis could be done after the completion of the game, so as not to slow down the interaction.
A first experiments has been made to transform the rule based decision tree into a classification model using a machine learning approach.
A first training made using a Naive Bayes Classifier shows encouraging results in discriminating between different classes of commands; the main difficulty here arises with the recognition of meaningless commands. In the current implementation our classifier will always choose a class, thus meaningless commands should be implemented in training like a class alongside the others, by providing a set of examples. This is clearly absurd, as anything can be a non-command. We are thus considering the possibility of setting a threshold for classifier confidence, and classify the decisions that fall below this level as 'unknown'[1].

5. Administrator's tools

The administrator's interface offers additional tools for drawing maps using a fixed set of landmarks and for drawing paths on maps. The administrator, typically the foreign language teacher or the second language acquisition researcher, can also access the logs of games played by users, which can form a research corpus of Italian as a second language.

6. Evaluation and use

Currently this tool is avalilable on the LIRA portal (http://elearning.unistrapg.it/firb/lira), which was created within the framework of an funded project dedicated to Italian as a Second and Heritage Language, and contains other facilities for language learning.
An evaluation is still ongoing, with first experiments assessing the usability of the interface, and will provide feedback for the improvement of the Natural Language Processing module.

7. Conclusions

In this abstract a prototype of computational infrastructure for human-machine map tasks has been presented. In the final poster/demo more details will be given on the architecture of the tool.
Map tasks can become a practical and reliable way for assessing communicative competence online and in an automated manner, furthering current attempts conducted within more traditional formats like multiple choice questions (e.g. Röver 2005). Furthermore, they are an ideal research ground in the fields of cognitive linguistics and artificial intelligence in order to investigate space orientation and man-machine interaction. From this point of view, one can investigate how learners perceive the machine as giver/follower in the task, what kind of perspective they tend to take (or imagine that the machine is taking) and what can be done in terms of simulating a more plausible joint interaction.

[1]We thank one of the anonymous reviewers of this paper for the suggestion.

8. Acknowledgements

The Map Task tool was jointly developed by ILC-CNR and the University of Modena and Reggio Emilia, and was funded under the national FIRB grant "Lingua-cultura Italiana In Rete per l'Apprendimento" (LIRA).

9. References

Anderson, A. et al. (1991). The HCRC Map Task Corpus. Language and Speech, 34, 351-366.

AVIP Project - Vocal Archives of Spoken Italian (n.d.); http://www.parlaritaliano.it

Bachman, L. (2002). Some reflections on task-based language performance assessment. Language Testing, 19, 453-476. Filipi, A., Wales, R. (2004). Perspective-taking and perspective-shifting associally situated and collaborative actions, Journal of Pragmatics [P], vol 36, Elsevier BV, Netherlands, pp. 1851-1884.

Gass, S. (1997) Input, interaction, and the second language learner. London: Routledge.

Gass, S., Mackey, A., Ross-Feldman, L. (2005). Task-Based Interactions in Classroom and Laboratory Settings, Language Learning, 55, 575-611

Gass, S., Mackey, A. (2007) Data elicitation for second and foreign language research, Routledge

Gilabert, R. Baron, J, Llanes, A. (2009). Manipulating cognitive complexity across task types and its impact on learners' interaction during oral performance. IRAL 47, 367-395 Interactive atlas of Catalan intonation (n.d.), http://prosodia.upf.edu/atlesentonacio/presentacio-english.html)

Kollar, T., Tellex, S., Roy, D., and Roy, N. (2010). "Toward Understanding Natural Language Directions", 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI): 259 - 266 .

Mackey, A. Gass S. (2006). Second Language Research: Methodology And Design. Lawrence Erlbaum Associates

Norris, J., Brown, J., Hudson, T., & Yoshioka, J. (1998). Designing second language performance assessments (Technical Report #18). Honolulu, HI: University of Hawai'i, Second Language Teaching & Curriculum Center.

Robinson, Peter (2001). Task complexity, task difficulty, and task production: Exploring interactions in a componential framework. Applied Linguistics 22 (1): 27-57.

Roventini, A. et al. (2003). ItalWordNet: Building a Large Semantic Database for the Automatic Treatment of Italian. In A. Zampolli, N. Calzolari, L. Cignoni, (Eds.) Linguistica Computazionale, vol. XVIII-XIX, Pisa-Roma, IEPI. Tomo II, pp. 745-791.

Röver, C. (2005) Testing ESL Pragmatics. Development and Validation of a Web-Based Assessment Battery Frankfurt: Peter Lang.