GRAPH: Realizing the Costs

Mariët Theune
University of Twente
The Netherlands
m.theune@utwente.nl

Jette Viethen
Macquarie University
Australia
jviethen@ics.mq.edu.au

Iris Hendrickx
University of Antwerp
Belgium
iris.hendrickx@ua.ac.be

Emiel Krahmer
Tilburg University
The Netherlands
e.j.krahmer@uvt.nl

Abstract

We describe the word string evaluation results obtained by coupling our graph-based attribute selection system with a simple realizer. Surprisingly, it turns out that in some cases higher attribute selection accuracy leads to larger differences between system-generated and human descriptions.

1 Introduction

In the context of the Referring Expression Generation (REG) 2008 Challenge, referring expressions are understood as *distinguishing descriptions*: descriptions that uniquely characterize a target object in a visual scene and do not apply to any of the other objects in the scene. Generating such descriptions is seen as a two-step procedure: (1) selecting a number of attributes of the target that characterize it uniquely, and (2) converting the selected set of attributes into a word string. In this paper we focus on step two, for which we use a simple template-based realiser written by Irene Langkilde-Geary from Brighton University. In last year's AS-GRE 2007 Challenge this realiser was used to produce surface realizations for human evaluation; see Belz and Gatt (2007). This year it was made available to all REG 2008 participants, to enable them to participate in Task 3 of the Challenge: 'end-to-end' referring expression generation. In the following, we first sketch how we obtained the input for realisation, and then we present the word string evaluation results, followed by a discussion.

2 Attribute selection: costs and orderings

As input for realisation, we use the attribute sets we generated for Task 1 (attribute selection) of REG 2008, using a version of the Graph-based REG algorithm of Krahmer et al. (2003). In the graph-based approach, objects and their attributes are represented as graphs, and generating referring expressions is seen as a graph search problem that outputs the cheapest distinguishing graph, given a particular *cost function* that assigns costs to attributes. By assigning zero costs to some attributes, the human tendency to mention redundant properties can be mimicked. As shown by Viethen et al. (2008), the *order* in which attributes are tried must also be controlled: if the graph search terminates before the free properties are tried, they are not included.

For Task 1 we investigated four cost functions: (1) **Simple**: all attributes have cost 1; (2) **Stochastic**: costs are based on the frequency of attributes in the REG 2008 training+development data; (3) **Free-Stochastic**: highly frequent attributes are free, while the others have stochastic costs; (4) **Free-Naive**: highly frequent attributes are free, while the others have a cost of 1 (somewhat infrequent) or 2 (very infrequent). These cost functions were combined with two property orderings: (A) **Random** and (B) **Cost-based**, where properties are tried in stochastic order from cheapest to most expensive. See Krahmer et al. (2008) for more details.

3 Word string evaluation results

We ran the Graph algorithm on the REG development data, using each of the cost + order combinations described above. Then we used the realizer to produce word strings expressing the generated attribute sets. The word strings were evaluated using two evaluation metrics: string-edit distance (EDIT, the Levenshtein distance between generated word string and human reference output, and string

Table 1: Results for the 2008 development set. The GRAPH 4+B settings were submitted to the REG 2008 Challenge.

Graph	Furniture			People			Overall		
System	EDIT	S-ACC	A-ACC	EDIT	S-ACC	A-ACC	EDIT	S-ACC	A-ACC
1+A	5.90	.04	.12	6.54	.00	.24	6.20	.02	.18
1+B	5.89	.04	.12	6.78	.00	.24	6.30	.02	.18
2+A	5.06	.05	.31	6.78	.00	.24	5.85	.03	.28
2+B	5.19	.05	.28	6.78	.00	.24	5.92	.03	.26
3+A	4.90	.05	.45	6.79	.00	.19	5.77	.03	.33
3+B	4.90	.05	.45	6.96	.00	.28	5.84	.03	.37
4+A	4.61	.05	.48	6.56	.00	.18	5.51	.03	.34
4+B	4.61	.05	.48	6.96	.00	.28	5.69	.03	.39

accuracy (S-ACC, the proportion of times the word string was identical to the reference. Table 2 summarizes the results. It also shows attribute accuracy (A-ACC): the proportion of times the generated attribute set was identical to the reference set. The table shows that in all cases, S-ACC is much lower than A-ACC. This is not surprising, since any given set of attributes can be expressed in many different (and equally good) ways, and the chance that the realizer produces exactly the same string as the human reference is quite small. For the furniture domain, we see that EDIT and S-ACC follow the same pattern as A-ACC: including redundant (free) properties leads to better scores (i.e., higher accuracy and smaller edit distance). However, for the people domain, we see that edit distance tends to get bigger as A-ACC increases, while S-ACC is 0 in all cases.

4 Discussion

To explain the evaluation results, we inspect those descriptions where A-ACC = 1 but S-ACC = 0, i.e., the attribute set is identical to the human reference but the word string is not. In setting 4+B (submitted to REG 2008) this is the case for 34 furniture and 19 people descriptions. For furniture, we see that the low S-ACC score can be largely explained by the fact that in 23 of the 34 descriptions the human reference either included no determiner or an indefinite one, whereas the system always included a definite determiner. In the people domain, the main reason for the zero string accuracy is the realizer's choice of "person" to express the type attribute, where the human references always have either "man" or "guy" (in line with the human preference for *basic level* values; cf. Krahmer et al. 2003). We also en-

counter the determiner problem again, aggravated by the fact that many person descriptions include embedded noun phrases (e.g., "man with beard").

To find out why edit distance increases (i.e., gets worse) as attribute accuracy increases for different system settings in the people domain, we look at the 6 descriptions that have A-ACC = 1 for setting 4+B but not 4+A. It turns out that 5 of the descriptions are realized as "the light-haired person with a beard", where the human reference strings are variations of "the man with a white beard", resulting in a relatively high edit distance. The problem here is that the link between beard and hair colour has got lost somewhere in the generation process. To achieve better results for both attribute selection and realization, such dependencies between attributes will clearly have to be taken into account.

References

Belz, A. and A. Gatt 2007. The attribute selection for GRE challenge: Overview and evaluation results *Proceedings of UCNLG+MT* 75-83.

Krahmer, E., S. van Erk and A. Verleg 2003. Graph-based generation of referring expressions. *Computational Linguistics, 29(1)*, 53-72.

Krahmer, E., M. Theune, J. Viethen, and I. Hendrickx 2008. GRAPH: The costs of redundancy in referring expressions. To appear in the *Proceedings of the* REG *Challenge 2008*, Salt Fork OH, USA.

Viethen, J., R. Dale, E. Krahmer, M. Theune and P. Touset. 2008. Controlling redundancy in referring expressions. *Proceedings LREC 08*, Marrakech, Morroco.