Evolutionary and Case-Based Approaches to REG: NIL-UCM-EvoTAP, NIL-UCM-ValuesCBR and NIL-UCM-EvoCBR

Raquel Hervás and Pablo Gervás
Natural Interaction based on Language (NIL)
Universidad Complutense de Madrid
raquelhb@fdi.ucm.es, pgervas@sip.ucm.es

1 Evolutionary Approach to Attribute Selection

We propose the use of evolutionary algorithms (EAs) (Holland, 1992) to deal with the attribute selection task of referring expression generation. Evolutionary algorithms operate over a population of individuals (possible solutions for a problem) that evolve according to selection rules and genetic operators. The fitness function is a metric that evaluates each of the possible solutions, ensuring that the average adaptation of the population increases each generation. Repeating this process hundreds or thousands of times leads to very good solutions for the problem.

We encode as a fitness function the specific constraints required for the reference to be acceptable. The crossover and mutation genetic operators ensure a reasonable variation between the different options much as a human-generated text would.

Each individual is represented by a set of genes that are the list of possible attributes in the reference. Each gene has an associated value of 0 (if the attribute is not included in the reference), or 1 (if the attribute is included in the reference). The initial population should have a low number of genes set to 1, because references tend to be short and the use of all the possible attributes should be avoided.

For the *crossover operator*, two individuals are selected randomly and crossed by a random point of their structure. For the *mutation operator*, some of the genes are chosen randomly to be mutated from 1 to 0, or vice versa.

The fitness function must find a balance between the univocal identification of a referent, and a natural use of attributes. The formula used as fitness function is defined in Equation 1:

$$fit_{ind_i} = f_{att_i} * weight_{att} + ident * weight_{id} \quad (1)$$

where *ident* represents whether the reference is univocally identifying the target among the distractors, and f_{att_i} computes the role of attributes as the normalised sum of the weight (depending on its absolute frecuency in ATTRIBUTE-SET elements in the corpus) of all attributes present (gene=1), as defined by Equation 2:

$$f_{att_i} = \frac{\sum gene_{att_i} * weight_{att_i}}{\#attsRef} \quad (2)$$

2 Case-Based Reasoning for Realization

Template-based solutions for natural language generation rely on reusing fragments of text extracted from typical texts in a given domain, applying a process of abstraction that identifies which part of them is common to all uses, and leaving certain gaps to be filled with details corresponding to a new use. A case-based solution (Aamodt and Plaza, 1994) to reference realization can obtain the information needed to realize a reference from the original examples of appropriate use that originated the templates.

In our approach, a case consists of a description of the problem (`ATTRIBUTE-SET`) and a solution (`ANNOTATED-WORD-STRING` interpreted as a template). Cases are stored in a Case Retrieval Net (CRN) (Lenz and Burkhard, 1996), a memory model developed to improve the efficiency of the retrieval tasks of the CBR cycle. Each attribute-value pair from the `ATTRIBUTE-SET` is a node in the net. Templates in `ANNOTATED-WORD-STRING` are considered as solutions to the cases. Similarities between the nodes are established for the retrieval stage of the CBR process. For example, we have considered that 'back' and 'right' orientation values have a higher similarity than 'back' and 'front' that are exactly the opposite.

The attribute-value pairs of `ATTRIBUTE-SET` that must be realized in a final string are used to query the net, which returns the more similar cases. Only one of them must be chosen to be adapted for the solution. We consider four different types of retrieved cases: *preferred* (cases with exactly the same attributes than the query), *more* (cases with the same attributes as the query and

Proceedings of the 12th European Workshop on Natural Language Generation, pages 187–188,
Athens, Greece, 30 – 31 March 2009. ©2009 Association for Computational Linguistics

		String Acc.	Edit Dist.	Norm. Edit Distance	BLEU 1 Score	BLEU 2 Score	BLEU 3 Score	BLEU 4 Score
EvoTAP	**Furniture**	0,08	4,87	0,51	0,44	0,33	0,24	0,18
	People	0,03	6,04	0,59	0,39	0,25	0,15	0,00
	Both	0,06	5,41	0,55	0,41	0,29	0,20	0,13
ValuesCBR	**Furniture**	0,01	5,91	0,55	0,44	0,31	0,20	0,13
	People	0,01	5,80	0,56	0,43	0,28	0,17	0,08
	Both	0,01	5,86	0,55	0,44	0,30	0,19	0,11
EvoCBR	**Furniture**	0,04	5,77	0,58	0,39	0,26	0,18	0,13
	People	0,01	6,94	0,61	0,41	0,25	0,16	0,08
	Both	0,03	6,31	0,59	0,41	0,26	0,17	0,11

Table 1: Results over development data for the three systems

some more), *lessExtra* (cases that lack some attribute from the query but have some extra ones), and *lessNoExtra* (cases that lack some attribute from the query and have no extra ones). The order given is the preferred order to chose the most suitable case for the query.

Adaptation of the chosen case depends on its type. The idea is to keep all the parts of the template that correspond to attributes common to the query and the case. Extra attributes in the case that do not appear in the query are discarded. Attributes in the query not appearing in the case are lost.

3 Results and Discussion

We have tested both solutions (evolutionary and case-based) separately and together in three different systems, relying on solutions presented in last year's challenge.

- **NIL-UCM-EvoTAP**. Selects attributes using the evolutionary solution and realises using the NIL-UCM-BSC solution (Gervás et al., 2008).
- **NIL-UCM-ValuesCBR**. Selects attributes using the NIL-UCM-MFVF solution (Gervás et al., 2008) and realizes using the case-based approach.
- **NIL-UCM-EvoCBR**. Selects attributes using the evolutionary solution and realizes using the case-based approach.

The results obtained by the three systems over development data are shown in Table 1.

The evolutionary approach performs poorly but might be improved by using a more refined algorithm for calculating attribute weights, such as done in the last year NIL-UCM-MFVF solution.

The reported CBR results were obtained over a case base built from a selection of the available training data (samples that relied on data not available in the input were omitted). This approach could be further refined by generating style-specific subsets of the case base.

Acknowledgments

This research is funded by the Spanish Ministry of Education and Science (TIN2006-14433-C02-01).

References

Aamodt, A. and Plaza, E.. 1994. Case-based reasoning: Foundational issues, methodological variations, and system approaches *AI Communications, 7(1).*

Gervás, P. and Hervás, R. and León, C. 2008. NIL-UCM: Most-Frequent-Value-First Attribute Selection and Best-Scoring-Choice Realization. *Referring Expression Generation Challenge 2008*, INGL-08, USA.

Holland, J.H. 1992. Adaptation in Natural and Artificial Systems. An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence. MIT Press, Cambridge, Massachusetts, Second Edition.

M. Lenz and H. Burkhard 1996. Case Retrieval Nets: Basic Ideas and Extensions. *Kunstliche Intelligenz.*