NIL: Attribute Selection for Matching the Task Corpus Using Relative Attribute Groupings Obtained from the Test Data

Raquel Hervás Pablo Gervás

Natural Interaction based on Language (NIL)
Universidad Complutense de Madrid
c/ Profesor José García Santesmases s/n, 28040 Madrid,
Spain
raquelhb@fdi.ucm.es, pgervas@sip.ucm.es

Abstract

The entry presented by the NIL (Natural Interaction based on Language) research group of the Universidad Complutense de Madrid was prepared by adapting existing software previously developed for a broader natural language generation application oriented towards the generation of fluent texts for storytelling. Only the identification, minimality and system-human match evaluation dimensions have been considered during development, and the final entry is specifically geared towards system-human match over the challenge corpus. The method employed for attribute selection is an adaptation of Reiter and Dale's fast efficient algorithm for referring expression generation, using relative groupings of attributes to determine the order in which they are considered. The relative groupings are obtained empirically from the training data. The results obtained over the development data indicate that the NIL entry is dealing adequately with issues of identification of referents. Results for system-human match are not so good, probably due to the fact that the corpus is a sample of possible references, rather than a selection of 'ideal' references.

Introduction

This document is a brief report on the entry for the first NLG Challenge on Attribute Selection for Referring Expressions Generation presented by the NIL (Natural Interaction based on Language) research group of the Universidad Complutense de Madrid. The entry was prepared by adapting existing software previously developed for a broader natural language generation application oriented towards the generation of fluent texts for storytelling. This document outlines the main characteristics of the challenge that were deemed relevant for the adaptation process, a brief sketch of the original software, a description of the adaptations carried out and how the training data were used in that process, and a report on the results obtained over the development data.

The TUNA Corpus (van Deemter et al., 2006) was designed to address the need for a data source to evaluate content determination in the Generation of Referring Expressions. This corpus contains human-authored descriptions that are paired with a domain representation listing all entities and their relevant attributes, and a semantic representation that maps parts of the linguistic description to the relevant attributes in the domain representation.
The corpus was collected using a web-based psycholinguistic experiment. Participants, who were self-reported native or fluent speakers of English, were shown trials consisting of one or two target referents and six distractors. Their task was to describe the target referents.
Data in the corpus was collected in two domains, one consisting of digitally constructed furniture/household items; another consisting of real photographs of people.

The NIL entry for the challenge was prepared based on TAP (Text Arranging Pipeline), an ongoing software development project. TAP is a set of interfaces that define generic functionality for a pipeline of tasks oriented towards natural language generation, from an initial conceptual input to surface realization as a string, with intervening stages of content planning and sentence planning. The approach to NLG used in TAP is very generic, since its original intended application was text generation for a storytelling application, and this must be able support varying degrees of complexity at the level of discourse structures, referring expressions, lexical choice, aggregation, and surface realization. However, most of the material available was not relevant for the purposes of the challenge, so the NIL entry has been built based on a single module of the pipeline: the Reference Solver.

Description of Our Method

The Reference Solver module is a submodule of the Sentence Planner module and it carries out the task of building appropriate referring expressions for referents described conceptually in a context given by the particular occurrence of a mention to the referent within a discourse. For the purpose of the challenge, a fragment of the code of the Reference Solver (that in charge of attribute selection) was isolated, and an interface was provided so that it might be fed with the relevant challenge data in the form of a target and a set of distractors. Such data are indeed handled in the original Reference Solver, together with additional material required to decide whether pronominal or onomastic references are to be used, and whether the reference should be definite or indefinite. The subtask of attribute selection in the Reference Solver is based on the algorithm described in (Reiter & Dale, 1992).

Adapting the TAP Reference Solver

Of the four evaluation dimensions considered in this challenge (identification, minimality, system-human match and task-based evaluation), only the first three have been used to adapt the Reference Solver module due to the fact that they can be computed automatically.
The initial Reference Solver module from TAP was implemented to generate always descriptions that univocally distinguish the target from the distractors in a

specific situation of the world. So, it was not necessary to modify the module to match the identification evaluation.
For the other two evaluation measures, the Reference Solver module was modified to accept not only a target and a set of distractors, but also a particular order in which to consider the attributes that are used in the process of generating the distinguishing expression. This corresponds to the list of *preferred attributes*, in order of preference, described in (Reiter & Dale, 1992). The attribute selection is completely dependant on the order in which the available attributes are considered.
In the case of the minimality evaluation, the attribute order was decided following the Full Brevity algorithm from Dale (1989). According to this algorithm, the list of attributes that are considered for the distinguishing description of the target can be ordered by their discriminatory power. Then, they are used in order until the target is univocally distinguished.
Finally, for the system-human match using the Dice coefficient, the order of the attributes is fixed by studying the training data as explained below.

Configuring the Reference Solver

To configure the modified module to deal with the challenge requirements, the training data was studied separately depending on the domain (furniture vs. people). Our idea was that not only the set of attributes in both domains was very different, but also that the psychological considerations taken into account for a person when referring to a piece of furniture or another person might be significantly different.
Following this trend, we considered the use of the *type* attribute as an additional distinguishing attribute when generating the descriptions in the furniture domain, but not in the people one (where all the elements involved are of type person).

Minimal Expression vs. Dice Coefficient
The first experiments carried out demonstrated that it was impossible to obtain good results simultaneously for the minimal and the system-human match evaluations. This is not surprising because human do not always use the optimal expression when referring to something, but they use a reference that is easier to formulate and/or understand.
In Table 1 we can see the results obtained (using the training data) for the Dice coefficient using the Reference Solver module adjusted to produce the minimal set of attributes for the expression. The percentage given by the Dice Calculator is quite small.

	Minimal	Dice
Furniture	100,00%	24,33%
People	100,00%	31,33%

Table 1: Results using the algorithm for the minimal expression for the training data

Considering these results, we decided to concentrate on improving the Dice coefficient results, not taking into account the order of consideration of the attributes that could produce the minimal reference.

Furniture Domain
Taking into account the previous considerations, in the furniture domain we had to work with a set of six attributes. All the possible combinations of them in different orders gave us 6! = 720 possibilities to explore. This is not much to be computed automatically, so we generated all the possible order combinations of the attributes and for each of them executed the whole process of generating the attribute selection corresponding to all the examples in the training corpus. The average of the Dice coefficient results was calculated in each case.
The study of these results revealed which combination of the attributes obtained the best results. But is also revealed a peculiarity of the way the quality of the results depended on the order of consideration of the attributes: it seemed to be dependant on the relative order in which certain 'groups' of attributes were considered, rather than the order of attributes in general. In other words, the results were almost the same for certain orders of groups of attributes, independently of the internal order inside these groups.
In the furniture domain the identified groups were [colour, type, size] and [orientation, x-dimension, y-dimension]. This distinction has some kind of psychological plausibility if we consider that one of the groups is more related with the spatial situation of the object, and the other with its own features. It seems possible that different people would feel more comfortable using one or another, depending on their general view of the world.
The best results obtained (using the training data) in the furniture domain are shown in Table 2. The order of the attributes was [type, colour, size, orientation, x-dimension, y-dimension].

	Minimal	Dice
Furniture	0,00%	82,45%

Table 2: Best results obtained in the furniture domain

People Domain
As mentioned before, in the people domain the type was not considered as a distinguishing attribute since it was always the same (person) for all referents considered. This leaves us with 11 attributes, and 11! = 39.916.800 possible orders for them, too many to be explored exhaustively.
Following the intuition about groupings of attributes obtained from the furniture domain, we carried out several experiments creating different combinations of the given attributes. Our first approach was to aggregate the attributes into three sets: [hasShirt, hasGlasses, hasSuit, hasTie] (clothing related things), [hasBeard, hairColour, hasHair, age] (appearance related things) and [x-dimension, y-dimension, orientation] (spatial situation), each of them containing attributes semantically related. Many combinations of the groups and the elements inside them were tested, but the results obtained with these divisions were not very good.
So, we tried another approach aggregating the attributes into groups depending on the relevance its presence or absence have to distinguish one person from another. For example, to have beard or to wear glasses are usually more perceivable than to wear a tie (especially if the person is also wearing suit). Four new groups were used

in the experiments: [hasSuit, hasTie, hasShirt], [hasBeard, hasGlasses, hasHair, hairColour], [age] and [x-dimension, y-dimension, orientation].
The best results obtained (using the training data) in the people domain are shown in Table 3. The order of the attributes used was [hasGlasses, hasBeard, hairColour, hasHair, hasSuit, hasTie, hasShirt, age, x-dimension, y-dimension, orientation].

	Minimal	Dice
People	42,72%	43,57%

Table 3: Best results obtained in the people domain

Minimal Expression vs. Dice Coefficient Revisited
The results obtained with the training data for the minimal expression evaluation deserve a few lines. In the case of the furniture domain, in general the values for the Dice coefficient were high, while the ones for the minimal expression were quite low. However, both results were very similar in most cases for the people domain.
Our impression is that this is due to the number of attributes considered in each domain. In the case of furniture, with only six attributes it is difficult for a person to find immediately the minimal reference between six distractors. However, in the people domain, with eleven attributes and only six distractors, it is likely that the target has one or two attributes that distinguish it from the rest of elements, and that they are easy to find for the person that is producing the reference.

Development Set Results

For the experiments over the development data we have used the combinations of attributes that showed better results with the training data. The results over the development data obtained using the final algorithm are given in Table 4. The results obtained using the modified module for the minimal reference are shown in Table 5.

	Identification	Minimal	Dice
Furniture	100%	0,00%	75,21%
People	100%	33,82%	44,78%

Table 4: Results obtained for the development data

	Minimal	Dice
Furniture	100,00%	20,95%
People	100,00%	30,93%

Table 5: Results using the algorithm for the minimal expression for the development data

As can be seen comparing Table 4 with the values obtained from the training data, there are no surprises in the final results.

Conclusions and Future Work

The results obtained over the development data indicate that the NIL entry is dealing adequately with issues of identification of referents. Minimal references might have been obtained with no problem (as shown in Table 5) had this been considered a priority. Another option that we did not follow might have been to attempt to improve the Dice coefficient results while maintaining the minimal reference. Dice coefficient results for our final method are rather poor when it comes to matching the specific expressions used in the corpus. We consider this to be due to the fact that the nature of the corpus indicates a broad variation in the type of expression used, aimed at describing a number of possible ways of describing referents as actually done by human evaluators, rather than setting the correct way of referring. Under these circumstances, we consider that to improve results in terms of similarity with a corpus would require an initial step of establishing a subcorpus of 'ideal references', and refining the software to obtain those.
As future work we have considered modelling particular styles of generating referring expressions, rather than working towards an ideal generic way. We could attempt this by narrowing the corpus to descriptions produced by a single person, and having the software model that person's particular way of generating descriptions. Such a solution would be useful in the context of a storytelling application, since it would provide the means of having different characters speaking with a different 'voice', or of having the same objects described in different ways when seen from the points of view of different characters.
One possible way of doing this might be to use a model of the speaker perception to guide the process of generating the referring expression. For instance, a professional of the fashion world might describe a person in terms of the clothes they are wearing whereas a doctor might rely more on their physical complexion. The technique we have used in configuring the NIL entry based on the training data relies on grouping the available attributes into subgroups that are considered always in a particular order. This technique might be refined by building a conceptual taxonomy of attributes, such that similar ones are classed under a more abstract concept and this information is used during the construction process to establish priorities between different attributes.

Acknowledgements

This research is funded by the Spanish Ministry of Education and Science (TIN2006-14433-C02-01 project) and a joint research group grant (UCM-CAM-910494) from the Universidad Complutense de Madrid and the Dirección General de Universidades e Investigación of the Comunidad Autónoma de Madrid.

Bibliographical References

Dale, R. (1989). Cooking Up Referring Expressions. In Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics, Vancouver BC, June.

van Deemter, K., van der Sluis, I. & Gatt, A. (2006). Building a semantically transparent corpus for the generation of referring expressions. Proceedings of the 4th International Conference on Natural Language Generation (Special Session on Data Sharing and Evaluation), INLG-06.

Reiter, E., Dale, R. (1992). A fast algorithm for the generation of referring expressions. Proceedings of the 14th conference on Computational linguistics, Nantes, France.