Graphs and Spatial Relations in the Generation of Referring Expressions

Jette Viethen
h.a.e.viethen@uvt.nl
TiCC
University of Tilburg
Tilburg, The Netherlands

Margaret Mitchell
m.mitchell@jhu.edu
HLT Centre of Excellence
Johns Hopkins University
Baltimore, USA

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

Abstract

When they introduced the Graph-Based Algorithm (GBA) for referring expression generation, Krahmer et al. (2003) flaunted the natural way in which it deals with relations between objects; but this feature has never been tested empirically. We fill this gap in this paper, exploring referring expression generation from the perspective of the GBA and focusing in particular on generating human-like expressions in visual scenes with spatial relations. We compare the original GBA against a variant that we introduce to better reflect human reference, and find that although the original GBA performs reasonably well, our new algorithm offers an even better match to human data (77.91% Dice). Further, it can be extended to capture speaker variation, reaching an 82.83% Dice overlap with human-produced expressions.

1 Introduction

Ten years ago, Krahmer et al. (2003) published the Graph-Based Algorithm (GBA) for referring expression generation (REG). REG has since become one of the most researched areas within Natural Language Generation, due in a large part to the central role it plays in communication: referring allows humans and language generation systems alike to invoke the entities that the discourse is about in the mind of a listener or reader.

Like most REG algorithms, the GBA is focussed on the task of selecting the semantic content for a referring expression, uniquely identifying a target referent among all objects in its visual or linguistic context. The framework used by the GBA is particularly attractive because it provides fine-grained control for finding the 'best' referring expression, encompassing several previous approaches. This control is made possible by defining a desired cost function over object properties to guide the construction of the output expression and using a search mechanism that does not stop at the first solution found.

One characteristic of the GBA particularly emphasized by Krahmer et al. (2003), advancing from research on algorithms such as the Incremental Algorithm (Dale and Reiter, 1995) and the Greedy Algorithm (Dale, 1989), was the treatment of relations between entities. Relations such as *on top of* or *to the left of* fall out naturally from the graph-based representation of the domain, a facet missing in earlier algorithms. We believe that this makes the GBA particularly well-suited for generating language in spatial visual domains.

In the years since the inception of the GBA, the REG community has become increasingly interested in evaluating algorithms against human-produced data in visual domains, aiming to mimic human references to objects. This interest has manifested most prominently in the 2007-2009 REG Challenges (Belz and Gatt, 2007; Gatt et al., 2008; Gatt et al., 2009) based on the TUNA Corpus (van Deemter et al., 2012). The GBA performed among the best algorithms in all three of these challenges. However, in particular its ability to analyze relational information could not be assessed, because the TUNA Corpus does not contain annotated relational descriptions.

We rectify this omission in the current work by testing the GBA on the GRE3D3 Corpus, which was designed to study the use of spatial relations in referring expressions (Viethen and Dale, 2008). We compare against a variant of the GBA that we introduce to build longer referring expres-

Proceedings of the 14th European Workshop on Natural Language Generation, pages 72–81,
Sofia, Bulgaria, August 8-9 2013.

sions, following the observation that humans tend to overspecify (i.e., not be maximally brief) in their referring expressions (Sonnenschein, 1985; Pechmann, 1989; Engelhardt et al., 2006; Arts et al., 2011). For both algorithms, we experiment with cost functions defined at different granularities to produce the best match to human data. We find that we can match human data better than the original GBA with the variant that encourages overspecification.

With this model, we aim to further advance towards human-like reference by developing a method to capture speaker-specific variation. Speaker variation cannot easily be modeled by the classic input variables of REG algorithms, but a number of authors have shown that system output can be improved by using speaker identity as an additional feature; this has often been accompanied by the observation that commonalities can be found in the reference behaviour of different speakers (Bohnet, 2008; Di Fabbrizio et al., 2008a; Mitchell et al., 2011b), particularly for spatial relations (Viethen and Dale, 2009). In the second experiment reported in this paper, we combine these insights by automatically clustering groups of speakers with similar behaviour and then defining separate cost functions for each group to better guide the algorithms.

Before we assess the ability of the GBA and our variant to produce human-like referring expressions containing relations (Sections 5 and 6), we will give an overview of the relevant background to the treatment of relations in REG, a short history of the GBA, and the relevance of individual variation (Section 2). We introduce our new variant graph-based algorithm, LongestFirst, in Section 3.

2 Relations, Graphs and Individual Variation

2.1 Relations in REG

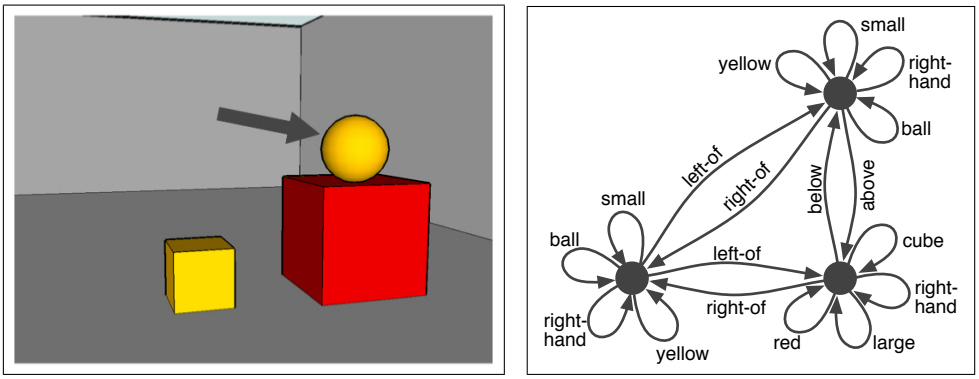
In the knowledge representation underlying most work in REG, each object in a scene is modeled as a set of attribute-value pairs describing the object's properties, such as $\langle \mathsf{size}, \mathsf{large} \rangle$. Such a representation is used in the two of the classic algorithms, the Greedy Algorithm (Dale, 1989) and the Incremental Algorithm (IA) (Dale and Reiter, 1995). Neither of these was originally intended to process relations between objects.

Several attempts have been made to adapt the traditional REG algorithms to include relations between objects in their output, but all of them suffer from problems with the knowledge representation not being suited to relations. Dale and Haddock (1991) use a constraint network and a recursive loop to extend the Greedy Algorithm, which uses the discriminatory power of an attribute as the main selection criterion. They treat relations the same as other attributes; but in most cases a certain spatial relation to a particular other object is fully distinguishing, which easily leads to strange chains of relations in the output omitting most other attributes (Viethen and Dale, 2006).

Krahmer and Theune (2002) suggest a similar adjustment for the IA by introducing a recursive loop if a relation to another object is introduced to the referring expression under construction. They treat relations as fundamentally different from other attributes in order to recognize when to enter the recursive loop, however, they fail to address the problem of infinite regress, whereby the objects in a domain might be described in a circular manner by the relations holding between them. Another relational extension to the IA has been proposed by Kelleher and Kruijff (2006), treating relations as a completely different class from other attributes. Both extensions of the IA make the simplifying assumption that relations should only be considered if it is not possible to fully distinguish the target referent from the surrounding objects in any other way, with the idea that it takes less effort to consider and describe only one object (Krahmer and Theune, 2002; Viethen and Dale, 2008).

2.2 A Short History of the GBA

A new approach to REG was proposed by Krahmer et al. (2003). In this approach, a scene is represented as a labeled directed graph (see Figure 1(b)), and content selection is a subgraph construction problem. Assuming a scene graph G = $\langle V_G, E_G \rangle$, where vertices V_G represent objects and edges E_G represent the properties and relations of these objects with associated costs, their algorithm returns the cheapest distinguishing subgraph that uniquely refers to the target object $v \in V_G$. Relations between objects (i.e., edges between different vertices) are a natural part of this representation, without requiring special computational mechanisms. In addition to cost functions, the GBA requires a preference ordering (PO) over the edges to arbitrate between equally cheap descriptions (Viethen et al., 2008).



(a) Scene 7 from the GRE3D3 Corpus. (b) A graph representing the scene to the left.

Figure 1: An example scene from the GRE3D3 Corpus and the corresponding domain graph.

As the cost functions and preference orders are specified over edges (i.e., properties), they allow much more fine-grained control over which properties to generate for a target referent than the attribute-based preference orders employed by the IA and its descendants. The cost functions can be used to give preference to a commonly used size value, such as large, over a rarely used color value, such as mauve, although in general color is described more often than size. This process is aided by a branch-and-bound search that guarantees to find the cheapest (i.e., 'best') referring expression.

Since its inception, the GBA has been shown to be useful for several referential phenomena. Krahmer and van der Sluis (2003) combined verbal descriptions with pointing gestures by modelling each such gesture as additional looping edges on all objects that it might be aimed at. While the authors confirmed the ideas implemented in the algorithm in psycholinguistic studies (van der Sluis, 2005), they never assessed its output in an actual domain.

van Deemter and Krahmer (2007) demonstrated how the GBA could be used to generate reference to sets as well as to negated and gradable properties by representing implicit information as explicit edges in domain graphs. They also presented a simple way to account for discourse salience based on restricting the distractor set. Its ability to cover such a breadth of referential phenomena makes the GBA a reasonably robust algorithm for further exploring the generation of human-like reference.

The GBA was systematically tested against human-produced referring expressions for the first time in the ASGRE Challenge 2007 (Belz and Gatt, 2007). This entry is described in detail in (Viethen et al., 2008) and was very successful as well in the following 2008 and 2009 REG Challenges (Gatt et al., 2008; Gatt et al., 2009) with a *free-naïve* cost function. This cost function assigns 0 cost to the most common attributes, 2 to the rarest, and 1 to all others. By making the most common attributes free, it became possible to include these attributes redundantly in a referring expression, even if they were not strictly necessary for identifying the target. The cost functions used in the challenges were attribute-based, and did therefore not make use of the refined control capabilities of the GBA.

Theune et al. (2011) used k-means clustering on the property frequencies in order to provide a more systematic method to transfer the FREE-NAÏVE cost function to new domains. They found that using only two clusters (a high frequency and a low frequency group with associated costs of 0 and 1) achieves the best results, with no significant differences to the FREE-NAÏVE cost function on the TUNA Corpus. Subsequently they showed that on this corpus, a training set of only 20 descriptions suffices to determine a 2-means cost function that performs as well as one based on 165 descriptions. In (Koolen et al., 2012), the same authors extended these experiments to a Dutch version of the TUNA Corpus (Koolen and Krahmer, 2010) and came to a similar conclusion. Neither of the corpora used in these experiments included relations between objects.

2.3 Individual Variation in REG

A number of authors have argued that to be able to produce human-like referring expressions, an algorithm must account for speaker variation: Different speakers will refer to the same object in different ways, and modeling this variation can bring us closer to generating the rich variety of ex-

pressions that people produce. Several approaches have been made in this direction.

Although this was not explicitly discussed in (Jordan and Walker, 2005), the machine-learned models presented there performed significantly better at replicating human-produced referring expressions when a feature set was used that included information about the identity of the speaker. In (Viethen and Dale, 2010), the impact of speaker identity as a machine-learning feature is more systematically tested. They show that exact knowledge about which speaker produced a referring expression boosts performance, but also find many commonalities between different speakers' strategies for content selection. Mitchell et al. (2011b) used participant identity in a machine learner to successfully predict the kind of size modifier to be used in a referring expression. Additionally, various submissions to the REG challenges, particularly by Bohnet and Fabbrizio et al. (Bohnet, 2008; Bohnet, 2009; Di Fabbrizio et al., 2008a; Di Fabbrizio et al., 2008b) used speaker-specific POs to increase performance in their adaptations of the IA.

All of these systems used the exact speaker identity as input, although many of the authors noted that groups of speakers behave similarly (Viethen and Dale, 2010; Mitchell et al., 2011b). We build off of this idea by clustering similar speakers together before learning parameters, and then generate for speaker-specific clusters. This method results in a significant improvement in performance.

3 LongestFirst: a New Search Strategy

The GBA guarantees to return the cheapest possible subgraph that fully distinguishes the target. However, many distinguishing subgraphs can have the same cost, for example, if a target can be identified either by its color or by its size, and color and size have the same cost. Viethen et al. (2008) discuss some examples in more detail.

In the case that more than one cheapest subgraph exists, the original GBA will generate the first it encountered. Due to its branch-and-bound search strategy, this is also the smallest subgraph, corresponding to the shortest possible description that can be found at the cheapest cost. Because its pruning mechanism does not allow further expansion of a graph once it is distinguishing, the number of attributes that the algorithm can include redundantly is limited, in particular if relations are involved. Attributes of visually salient nearby landmark objects that are introduced to the referring subgraph by a relation are only considered after all other attributes of the target object. This is the case even if these attributes are free and feature early in the preference order.

The GBA is therefore not able to replicate many overspecified descriptions that human speakers may use: if a subgraph containing a relation is already distinguishing before the attributes of a landmark object are considered, the algorithm will not include any information about the landmark. Not only is it unlikely that a landmark object should be included in a description without any further information about it, it also seems intuitive that speakers with a preference for certain attributes (such as color) would include these attributes not only for the target referent, but for a landmark object as well.

We solve this problem by amending the search algorithm in a way that finds the *longest* of all the cheapest subgraphs, and call the resulting algorithm *LongestFirst*. This search strategy results in a much larger number of subgraphs to check, in particular, when used with cost functions that involve a lot of free edges. In order to keep our systems tractable, we therefore limit the number of attributes the LongestFirst algorithm can include to four, based on the finding from (Mitchell et al., 2011a) that people rarely include more than four modifiers in a noun phrase. In Experiment 2 we additionally test a setting in which the maximum number of attributes is determined on the basis of the average description length in the training data.

4 Implementation Note

The original implementation of the GBA did not provide a method to specify the order in which edges were tried, although the edge order determines the order in which distinguishing subgraphs are found by the algorithm (Krahmer et al., 2003). This was fixed in (Viethen et al., 2008) by adding a PO as parameter to the GBA to arbitrate between equally cheap solutions.

A further issue arose in this implementation when tested on the GRE3D3 domain, because there was no simple way to specify which object each property belonged to; for the TUNA domain where the GBA has traditionally been evaluated, it is safe to always assume a property belongs to the

target referent. We have therefore provided additional functionality to the GBA that requires that not only ⟨attribute, value⟩ pairs are specified, but ⟨entity1, attribute, value, entity2⟩ tuples, which can be translated directly into graph edges. For example the tuple ⟨tg:relation:above:lm⟩ represents the edge labelled above between the yellow ball and the red cube in Figure 1. For direct attributes, such as size or color, entity1 and entity2 in these tuples are identical, resulting in loop edges. This Java implementation of the GBA and the Python implementation of the LongestFirst algorithm are available at `www.m-mitchell.com/code`.

5 Experiment 1: Relational Descriptions

In our first experiment, we evaluate how well the GBA produces human-like reference in a corpus that uses spatial relations. We compare against the LongestFirst variant that encourages overspecification.

5.1 Material

To evaluate the different systems, we use the GRE3D3 Corpus. It consists of 630 distinguishing descriptions for objects in simple 3D scenes. Each of the 20 scenes contains three objects in different spatial relations relative to one another (see Figure 1). The target referent, marked by an arrow, was always in a direct adjacency relation ($\mathsf{on-top-of}$ or $\mathsf{in-front-of}$) to one of the other two objects, while the third object was always placed at a small distance to the left or right. The objects are either spheres or cubes and differ in size and color. In addition to these attributes, the 63 human participants who contributed to the corpus used the objects' location as well as the spatial relation between the target referent and the closest landmark object. Each participant described one of two sets of 10 scenes. The scenes in the two sets are not identical, but equivalent, so the sets can be conflated for most analyses. Spatial relations were used in 36.6% (232) of the descriptions, although they were never necessary to distinguish the target object. Further details about the corpus may be found in (Viethen and Dale, 2008).

5.2 Approaches to Parameter Settings

As discussed above, the GBA behaves differently depending on the PO and the cost functions over its edges. To find the best match with human data, we explore several different approaches to setting these two parameters. An important distinction between the approaches we try hinges on the difference between *attributes* and *properties*. Attributes correspond to, e.g., color, size, or location, while properties are attribute-value pairs, e.g., ⟨color, red⟩, ⟨size, large⟩, ⟨location, middle⟩.

Previous evaluations of the GBA typically used parameter settings based on either attribute frequency (Viethen et al., 2008) or property frequency (Koolen et al., 2012). We compare both methods for setting the parameters. Because the scenes on which the corpus is based were not balanced for the different attribute-values, the frequency of a property is calculated as the proportion of descriptions in which it was used for those scenes where the target actually possessed this property. For our evaluation, the trainable costs and the POs are determined using cross-validation (see Section 5.3). We use the following approaches:

0-COST-PROP: All edges have 0 cost, and the PO is based on property frequency. Each property is included (regardless of how distinguishing it is) until a distinguishing subgraph is found.

0-COST-ATT: As 0-COST-PROP, but the PO is based on attribute frequency.

FREE-NAÏVE-PROP: Properties that occur in more than 75% of descriptions where they could be used cost 0, properties with a frequency below 20% cost 2, and all others cost 1 (Viethen et al., 2008). The PO is based on property frequency.

FREE-NAÏVE-ATT: As FREE-NAÏVE-PROP:, but costs and PO are based on attribute frequency.

K-PROP: Costs are assigned using k-means clustering over property frequencies with k=2 (Theune et al., 2011). The PO is based on property frequency.

K-ATT: As K-PROP, but the k-means clustering and the PO are based on attribute frequency.

5.3 Evaluation Setup

We evaluate the version of the GBA used by Viethen et al. (2008), with additional handling for relations between entities (see Section 4). We compare against our LongestFirst algorithm from Section 3 on all approaches described in Section 5.2. As baselines, we compare against the Incremental Algorithm (Dale and Reiter, 1995) and a simple informed approach that includes attributes/properties seen in more than 50% of the

training descriptions. We do not use the IA's relational extensions (Krahmer and Theune, 2002; Kelleher and Kruijff, 2006), because these would deliver the same relation-free output as the basic IA (relations are never necessary for identifying the target in GRE3D3). These two baselines are tried with an attribute-based PO and a property-based one. We do not expect a difference between the attribute- and the property-based PO on the IA, as this difference would only come to the fore in a situation where a choice has to be made between two values of the same attribute. In the IA's analysis of the GRE3D3 domain, this can only happen with relations, which it will not use in this domain.

We use Accuracy and Dice, the two most common metrics for human-likeness in REG (Gatt and Belz, 2008; Gatt et al., 2009), to assess our systems. Accuracy reports the relative frequency with which the generated attribute set and the human-produced attribute set match exactly. Dice measures the overlap between the two attribute sets. For details, see, for example, Krahmer and van Deemter's (2012) survey paper. We train and test our systems using 10-fold cross-validation.

5.4 Results

The original version of the Graph-Based Algorithm shows identical performance for all approaches (See Table 1). All use a preference order starting with type, followed by color and size, and a cost function that favors the same attributes. As these attributes always suffice to distinguish the intended referent, the algorithm stops before spatial relations are considered. For the scene in Figure 1 it includes the minimal content ⟨tg:type:ball⟩, but for a number of scenes it overspecifies the description.

The LongestFirst/0-COST systems and the LongestFirst/K-PROP system are the only systems that include relations in their output. The LongestFirst/0-COST systems both include a relation in every description; however, not always the one that was included in the human-produced reference, resulting in 521 false-positives for the attribute-based version and 398 for the property-based one. For the scene in Figure 1 they include ⟨tg:color:yellow, tg:size:small, tg:type:ball, tg:right_of:obj3⟩ and ⟨tg:color:yellow, tg:size:small, tg:type:ball, tg:on_top_of:lm⟩, respectively. The first one of these two attribute sets (produced by

		Original GBA	Longest First
0-COST-PROP	Acc	39.21	0.16
	Dice	73.40	68.75
0-COST-ATT	Acc	39.21	0.00
	Dice	73.40	64.34
FREE-NAÏVE-ATT	Acc	39.21	**46.51**
	Dice	73.40	**77.91**
FREE-NAÏVE-PROP	Acc	39.21	38.10
	Dice	73.40	74.99
K-PROP	Acc	39.21	35.08
	Dice	73.40	74.66
K-ATT	Acc	39.21	35.08
	Dice	73.40	74.56
		50%-Base	IA
prop-based PO	Acc	27.30	37.14
	Dice	72.17	72.21
att-based PO	Acc	24.92	37.14
	Dice	71.16	72.21

Table 1: Experiment 1: System performance in %. We used χ^2 on Accuracy and paired t-tests on Dice to check for statistical significance. The best performance is highlighted in boldface. It is statistically significantly different from all other systems (Acc: $p < 0.02$, Dice: $p < 0.0001$).

LongestFirst/0-COST-ATT) includes the relation between the target and the third object to the right, which was almost never included in the human-produced references, leading to many false-positives. The LongestFirst/K-PROP system results in only 45 true-positives and 81 false-positives. It includes the attribute set ⟨tg:color:yellow, tg:type:ball⟩ for Figure 1. One of its relational descriptions (for Scene 5) contains the set ⟨tg:size:small, tg:color:blue, tg:on_top_of:lm⟩.

The 50%-baseline system outperforms the LongestFirst/0-COST systems, which illustrates the utility of cost functions in combination with a PO. It includes the attribute set ⟨tg:color:yellow, tg:type:ball⟩ for the scene in Figure 1. The best performing system is the LongestFirst algorithm with the attribute-based FREE-NAÏVE approach, although this system produces no spatial relations.

6 Experiment 2: Individual Variation

We now extend our methods to take into account individual variation in the content selection for referring expressions, and evaluate whether we have better success at reproducing participants' relational descriptions. Rather than using speaker identity as an input parameter to the system (Section 2.3), we automatically find groups of people

who behave similarly to each other, but significantly different to speakers in the other groups.

6.1 Evaluation Setup

We use k-means clustering to group the speakers in the GRE3D3 Corpus based on the number of times they used each attribute and the average length of their descriptions. We tried values between 2 and 5 for k, but found that any value above 2 resulted in two very large clusters accompanied by a number of extremely small clusters. As these small clusters would not be suitable for x-fold cross-validation, we proceed with two clusters, one consisting of speakers preferring relatively long descriptions that often contain spatial relations (Cluster CL0, 16 speakers, 160 descriptions), and one consisting of speakers preferring short, non-relational descriptions (Cluster CL1, 47 speakers, 470 descriptions).

We train cost functions and POs separately for the two clusters in order to capture the different behaviour patterns they are based on. We use the FREE-NAÏVE cost functions for this experiment, which outperformed all others in Experiment 1. We again use 10-fold cross-validation for the evaluation. In this experiment, we vary the maximum length setting for the LongestFirst algorithm. In Experiment 1, the maximum length for a referring expression was set to 4 based on previous empirical findings. Here we additionally test setting it to the rounded average length for each training fold. On Cluster CL0 this average length is 6 in all folds, on Cluster CL1 it is 3.

6.2 Results

As shown in Table 2, the LongestFirst algorithm performs best at generating human-like spatial relations (Cluster CL0), with property-based parameters and a maximum description length determined by the training set. It produces the attribute set ⟨lm:type:cube, tg:on_top_of:lm, tg:type:ball, tgcolouryellow, lm:colour:red⟩ for Figure 1. The difference to the other systems is statistically significant for both Accuracy (χ^2>15, p<0.0001) and Dice (t>13, p<0.0001). The attribute-based parameters and the original GBA perform very badly on this cluster. For participants who do not tend to use spatial relations (Cluster CL1), the maximum length setting has no influence, but attribute-based parameters perform better than property-based ones. The attribute-based LongestFirst systems also outperform the original GBA on CL1, but interestingly none of the differences are as large as on CL0. For the scene in Figure 1 they produce the attribute set ⟨tg:type:ball, tg:colour:yellow⟩.

			CL0	CL1	avg
LongestFirst -max-av	FN -PROP	Acc	**19.38**	48.94	**41.43**
		Dice	**75.61**	80.27	**79.08**
	FN -ATT	Acc	0.00	**60.00**	**44.76**
		Dice	55.74	**85.28**	77.78
LongestFirst -max4	FN -PROP	Acc	0.63	48.94	36.67
		Dice	72.15	80.21	78.17
	FN -ATT	Acc	0.00	**60.00**	**44.76**
		Dice	59.01	**85.28**	**78.61**
Original GBA	FN -PROP	Acc	5.00	48.30	37.30
		Dice	49.36	80.77	72.79
	FN -ATT	Acc	5.00	50.85	**39.21**
		Dice	49.36	81.58	73.40

Table 2: Experiment 2: Performance in % of the LongestFirst and OriginalGraph algorithms on the two speaker clusters and overall using the FREE-NAÏVE (FN) approaches. We used χ^2 on Accuracy and paired t-tests on Dice to check for statistical significance. The best performance in each column and those that are statistically not significantly different are highlighted in boldface.

The average results over both clusters (shown in the last column Table 2) are not conclusive as to which setting should be used overall, although it is clear that the LongestFirst version is preferable when evaluated by Dice. The different result patterns on the two clusters suggest that the different referential behaviour of the participants in the two clusters are ideally modeled using different parameters. In particular, it appears that *property*-based costs are useful for replicating descriptions containing relations to other objects, while *attribute*-based costs are useful for replicating shorter descriptions. The best overall performance, achieved by combining the best performing systems on each cluster (LongestFirst-max-av/FN-PROP on CL0 and LongestFirst/FN-ATT with either maximum length setting on CL1), lies at 49.68% Accuracy and 82.83% Dice. The Dice score in this combined model is significantly higher than the best achieved by LongestFirst-max-av/FN-PROP and from the best Dice score achieved on the unclustered data in Experiment 1 (t=8.2, p<0.0001). The difference in Accuracy is not significant (χ^2=1.2, p> 0.2).

To get an idea of how successful the new LongestFirst approach is at replicating the use of relations on the clustered data, we take a closer look at the output of the best-performing systems

on the two clusters. On CL0, the cluster of participants who produce longer descriptions containing more spatial relations, the best match to the human data comes from LongestFirst-max-av/FN-PROP. 147 of the 160 descriptions in this cluster contain a relation, and the system includes the correct relation for all 147. It falsely also includes a relation for the remaining 13 descriptions. This shows that with the appropriate parameter settings the LongestFirst algorithm is able to replicate human relational reference behaviour, but personal speaker preferences are the main driving factor for the human use of relations.

CL1, the cluster with shorter descriptions, contains only 85 (18%) relational descriptions. The best performing system on this cluster (LongestFirst/FN-ATT) does not produce any relations. This is not surprising as the cost functions and POs for this cluster are necessarily dominated by the non-relational attributes used more regularly. The cases in which relations are used stem from participants who do not show a clear preference for or against relations and would therefore be hard to model in any system. With more data it might be possible to group these participants into a third cluster and find suitable parameter settings for them. This would only be possible if their use of relations is influenced by other factors available to the algorithm, such as the spatial configuration of the scene. Viethen and Dale's (2008) analysis of the GRE3D3 Corpus suggests that this is the case at least to some extent.

7 Conclusions and Future Work

We have evaluated the Graph-Based Algorithm for REG (Krahmer et al., 2003) as well as a novel search algorithm, LongestFirst, that functions on the same graph-based representation, to assess their ability to generate referring expressions that contain spatial relations. We coupled the search algorithms with a number of different approaches to setting the cost functions and preference orders that guide the search.

In Experiment 1, we found that ignoring the cost function (our 0-cost approaches) is not helpful; but the LongestFirst algorithm, which produces longer descriptions, leads to more human-like output for the visuospatial domain we evaluate on than the original Graph-Based Algorithm or the Incremental Algorithm (Dale and Reiter, 1995). However, in order for spatial relations to be included in a human-like way, it was necessary to take into account speaker preferences. We modeled these in Experiment 2 by clustering the participants who had contributed to the evaluation corpus based on their referential behaviour. By training separate cost functions and preference orders for the different clusters, we enabled the LongestFirst algorithm to correctly reproduce 100% of relations used by people who regularly mentioned relations.

Our findings suggest that the graph-based representation proposed by Krahmer et al. (2003) can be used to successfully generate relational descriptions, however their original search algorithm needs to be amended to allow more overspecification. Furthermore, we have shown that variation in the referential behaviour of individual speakers has to be taken into account in order to successfully model the use of relations in referring expressions. We have proposed a clustering approach to advance this goal based directly on the referring behaviour of speakers rather than speaker identity. We have found that the best models use fine-grained property-based parameters for speakers who tend to use spatial relations, and coarser attribute-based parameters for speakers who tend to use shorter descriptions.

In future work, we hope to expand to more complex domains, beyond the simple properties available in the GRE3D3 Corpus. We also aim to explore further graph-based representations and search strategies, modeling non-spatial properties as separate vertices, similar to the approach by Croitoru and van Deemter (2007).

8 Acknowledgements

Viethen and Krahmer received financial support from The Netherlands Organization for Scientific Research, (NWO, Vici grant 277-70-007), and Mitchell received financial support from the Scottish Informatics and Computer Science Alliance (SICSA), which is gratefully acknowledged.

References

Anja Arts, Alfons Maes, Leonard Noordman, and Carel Jansen. 2011. Overspecification in written instruction. *Linguistics*, 49(3):555–574.

Anja Belz and Albert Gatt. 2007. The Attribute Selection for GRE Challenge: Overview and evaluation results. In *Proceedings of the Workshop on Using Corpora for NLG: Language Generation and*

Machine Translation (UCNLG+MT), pages 75–83, Copenhagen, Denmark.

Bernd Bohnet. 2008. The fingerprint of human referring expressions and their surface realization with graph transducers. In *Proceedings of the 5th International Conference on Natural Language Generation*, pages 207–210, Salt Fork OH, USA.

Bernd Bohnet. 2009. Generation of referring expression with an individual imprint. In *Proceedings of the 12th European Workshop on Natural Language Generation*, pages 185–186, Athens, Greece.

Madalina Croitoru and Kees van Deemter. 2007. A conceptual graph approach to the generation of referring expressions. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 2456–2461, Hyderabad, India.

Robert Dale and Nicholas Haddock. 1991. Generating referring expressions involving relations. In *Proceedings of the 5th Conference of the European Chapter of the Association for Computational Linguistics*, pages 161–166, Berlin, Germany.

Robert Dale and Ehud Reiter. 1995. Computational interpretations of the Gricean maxims in the generation of referring expressions. *Cognitive Science*, 19(2):233–263.

Robert Dale. 1989. Cooking up referring expressions. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 68–75, Vancouver BC, Canada.

Giuseppe Di Fabbrizio, Amanda Stent, and Srinivas Bangalore. 2008a. Referring expression generation using speaker-based attribute selection and trainable realization (ATTR). In *Proceedings of the 5th International Conference on Natural Language Generation*, pages 211–214, Salt Fork OH, USA.

Giuseppe Di Fabbrizio, Amanda J. Stent, and Srinivas Bangalore. 2008b. Referring expression generation using speaker-based attribute selection and trainable realization. In *Twelfth Conference on Computational Natural Language Learning*, Manchester, UK.

Paul E. Engelhardt, Karl D. Bailey, and Fernanda Ferreira. 2006. Do speakers and listeners observe the gricean maxim of quantity? *Journal of Memory and Language*, 54:554–573.

Albert Gatt and Anja Belz. 2008. Attribute selection for referring expression generation: New algorithms and evaluation methods. In *Proceedings of the 5th International Conference on Natural Language Generation*, pages 50–58, Salt Fork OH, USA.

Albert Gatt, Anja Belz, and Eric Kow. 2008. The TUNA Challenge 2008: Overview and evaluation results. In *Proceedings of the 5th International Conference on Natural Language Generation*, pages 198–206, Salt Fork OH, USA.

Albert Gatt, Anja Belz, and Eric Kow. 2009. The TUNA-REG Challenge 2009: Overview and evaluation results. In *Proceedings of the 12th European Workshop on Natural Language Generation*, pages 174–182, Athens, Greece.

Pamela W. Jordan and Marilyn Walker. 2005. Learning content selection rules for generating object descriptions in dialogue. *Journal of Artificial Intelligence Research*, 24:157–194.

John Kelleher and Geert-Jan Kruijff. 2006. Incremental generation of spatial referring expressions in situated dialog. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 1041–1048, Sydney, Australia.

Ruud Koolen and Emiel Krahmer. 2010. The D-TUNA Corpus: A dutch dataset for the evaluation of referring expression generation algorithms. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*, Valetta, Malta.

Ruud Koolen, Emiel Krahmer, and Mariët Theune. 2012. Learning preferences for referring expression generation: Effects of domain, language and algorithm. In *Proceedings of the 7th International Natural Language Generation Conference*, pages 3–11, Starved Rock, IL, USA.

Emiel Krahmer and Mariët Theune. 2002. Efficient context-sensitive generation of referring expressions. In Kees van Deemter and Rodger Kibble, editors, *Information Sharing: Reference and Presupposition in Language Generation and Interpretation*, pages 223–264. CSLI Publications, Stanford CA, USA.

Emiel Krahmer and Kees van Deemter. 2012. Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.

Emiel Krahmer and Ielka van der Sluis. 2003. A new model for generating multimodal referring expressions. In *Proceedings of the 9th European Workshop on Natural Language Generation*, pages 47–57, Budapest, Hungary.

Emiel Krahmer, Sebastiaan van Erk, and André Verleg. 2003. Graph-based generation of referring expressions. *Computational Linguistics*, 29(1):53–72.

Margaret Mitchell, Aaron Dunlop, and Brian Roark. 2011a. Semi-supervised modeling for prenominal modifier ordering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 236–241, Portland OR, USA.

Margaret Mitchell, Kees van Deemter, and Ehud Reiter. 2011b. Applying machine learning to the choice of size modifiers. In *Proceedings of the 2nd Workshop on the Production of Referring Expressions*, Boston MA, USA.

Thomas Pechmann. 1989. Incremental speech production and referential overspecification. *Linguistics*, 27:89–110.

Susan Sonnenschein. 1985. The development of referential communication skills: Some situations in which speakers give redundant messages. *Journal of Psycholinguistic Research*, 14(5):489–508.

Mariët Theune, Ruud Koolen, Emiel Krahmer, and Sander Wubben. 2011. Does size matter - how much data is required to train a REG algorithm? In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 660–664, Portland OR, USA.

Kees van Deemter and Emiel Krahmer. 2007. Graphs and Booleans: On the generation of referring expressions. In Harry C. Bunt and Reinhard Muskens, editors, *Computing Meaning*, volume 3, pages 397–422. Kluwer, Dordrecht, The Netherlands.

Kees van Deemter, Albert Gatt, Ielka van der Sluis, and Richard Power. 2012. Generation of referring expressions: Assessing the incremental algorithm. *Cognitive Science*, 36(5):799–836.

Ielka van der Sluis. 2005. *Multimodal Reference, Studies in Automatic Generation of Multimodal Referring Expressions*. Ph.D. thesis, Tilburg University, The Netherlands.

Jette Viethen and Robert Dale. 2006. Algorithms for generating referring expressions: Do they do what people do? In *Proceedings of the 4th International Conference on Natural Language Generation*, pages 63–70, Sydney, Australia.

Jette Viethen and Robert Dale. 2008. The use of spatial relations in referring expression generation. In *Proceedings of the 5th International Conference on Natural Language Generation*, pages 59–67, Salt Fork OH, USA.

Jette Viethen and Robert Dale. 2009. Referring expression generation: What can we learn from human data? In *Proceedings of the 2009 Workshop on Production of Referring Expressions: Bridging the Gap Between Computational and Empirical Approaches to Reference*, Amsterdam, The Netherlands.

Jette Viethen and Robert Dale. 2010. Speaker-dependent variation in content selection for referring expression generation. In *Proceedings of the 8th Australasian Language Technology Workshop*, pages 81–89, Melbourne, Australia.

Jette Viethen, Robert Dale, Emiel Krahmer, Mariët Theune, and Pascal Touset. 2008. Controlling redundancy in referring expressions. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, Marrakech, Morocco.