Natural Language Directed Inference in the Presentation of Ontologies

Chris Mellish and Xiantang Sun
Department of Computing Science
University of Aberdeen
Aberdeen AB24 3UE, UK
{cmellish,xsun}@csd.abdn.ac.uk

Abstract

It is hard to come up with a general formalisation of the problem of *content determination* in natural language generation because of the degree of domain-dependence that is involved. This paper presents a novel way of looking at a class of content determination problems in terms of a non-standard kind of inference, which we call *natural language directed inference*. This is illustrated through examples from a system under development to present parts of ontologies in natural language. Natural language directed inference represents an interesting challenge to research in automated reasoning and natural language processing.

1 Introduction: Content Determination in NLG

Content determination in natural language generation (NLG) is the task of determining relevant material to be included in a natural language text. Usually this is taken to involve in addition some planning of the overall structure of the text, as this will affect whether particular combinations of content will be able to be coherently realised. Because the details of content determination depend on characteristics of the application domain and in general every NLG system has a different domain or goals, there has been little success in coming up with general models of the structure of this process. In particular, reference architectures for NLG have relatively little to say about it [Mellish *et al.*, 2004].

It is useful to distinguish between two broad classes of content determination problems. "Top-down" problems have specific goals in terms, for instance, of convincing or persuading the reader about something. These have typically been addressed through the framework of planning (e.g. [Moore, 1994]), where content is sought to fill in requirements of planning operators. Here the requirement to build a successful argument of some kind drives the process. On the other hand, "bottom-up" problems require the production of a more general expository or descriptive text that puts together information to satisfy more diffuse goals. For these problems text coherence is more imporant than which particular arguments or points are made. For instance, the ILEX project aimed to emulate a museum curator telling a good story to link together a sequence of selected exhibits. It was argued that a more opportunistic approach to content determination was needed for this sort of application [Mellish *et al.*, 1998].

In this paper, we concentrate primarily on the "bottom-up" type of content determination problem. But what makes content determination hard in either case is largely the fact that two different "worlds" are involved – the domain model and the linguistic world. Content determination is selecting material from the (not necessarily very linguistic) domain model, e.g. facts, rules and numbers, in the hope that it will permit a coherent realisation as a text. In between the domain model Θ and the set of possible texts $Text$ sits a possibly non-trivial mapping ρ ("realisation"):

$$\rho : \{\theta | \theta \text{ is content selected from } \Theta\} \rightarrow 2^{Text}$$

The problem is that since ρ may be complex, it will be hard to judge which content will yield the most successful text. Meteer [1992] pointed out that this "generation gap" in the worst case will mean that content is formulated which is not expressible in language at all. This is also related to the "problem of logical form equivalence" [Shieber, 1993] which arises because from a domain point of view two logically equivalent formulae are interchangeable and so it is a matter of chance which of the many logically equivalent formulae is given to a realiser. ρ must therefore be able to choose between realisations corresponding to all formulae logically equivalent to its input.

The problems raised by Meteer and Shieber do not, however, always arise in practice. In many applications, the possible forms of θ are restricted enough and close enough to a linguistic representation that one can be sure that there will always be at least one value for ρ. Also ρ doesn't have to map onto all possible texts – one can artificially limit the extent to which realisation diverges from what is suggested by the surface form of the input. All NLG systems adopt these sorts of simplications.

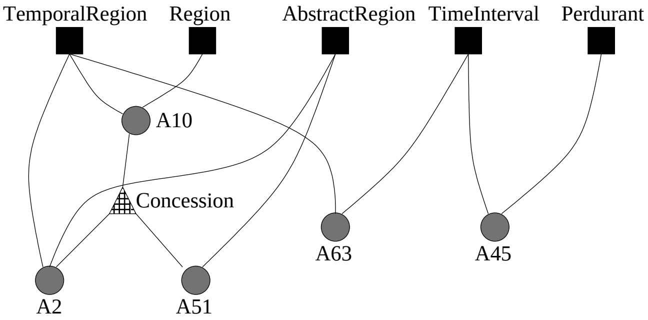
In the next two sections, we show how realisation and content determination initially worked in our project to present ontologies in natural language. In section 4 we then consider limitations of this approach to content determination, which gives rise to the novel idea of treating content determination as a kind of inference, *natural language directed inference*. We then outline our initial steps to implement this process and how it relates to existing work in automated reasoning and natural language generation.

2 Realisation from Ontology Axioms

Our current research addresses the problem of presenting parts of OWL DL [McGuinness and van Harmelen, 2004] ontologies in natural language. This will extend existing approaches to generating from simpler DLs (e.g. [Wagner *et al.*, 1999]) by taking into account the fact that in a language like OWL DL a concept is described more by a set of constraints than by a frame-like definition. For instance, the bottom of Figure 1 shows a set of axioms relevant to the concept $TemporalRegion$ in an example ontology. Because there may be a number of axioms providing different facts about a concept, the information cannot in general be presented in a single sentence but requires an extended text with multiple sentences, the overall structure having to be planned so as to be coherent as a discourse. Our work is also different from other work which generates text about *individuals* described using ontologies [Wilcock, 2003; Bontcheva and Wilks, 2004], in that it presents the ontology *class axioms* themselves.

In this section, we give an example of how ρ complicates the reasoning about appropriate content, by showing that although the ρ that we are developing is relatively simple, it nevertheless complicates decisions about the complexity of what can be presented in a sentence.

Given an axiom to be expressed as a sentence (we discuss in section 3 how such axioms are selected), our realisation approach uses rules with simple recursive structural patterns and assembles text with grammatically-annotated templates[1]. The idea is that we will collect rules for special case expressions that can be realised relatively elegantly in language as well as having generic rules that ensure that every possible structure can be handled. Optimal English will arise from detecting the part of speech of any class and role names which are English words (as well as cases such as multiple word names and roles such as “hasX”, “Xof” where X is a noun), and we have been able to obtain this information with reasonable quality automatically using WordNet[2]. Unless such conventions are used in the ontology definition or the reader is familiar with some of the ontology terms, it will not be possible to convey any useful information to them without extra domain-specific resources.



A10: $TemporalRegion \sqsubseteq Region$
A2: $AbstractRegion \sqcap TemporalRegion = \perp$
A63: $TimeInterval \sqsubseteq TemporalRegion$
A45: $Perdurant \sqsubseteq \forall HappenAt.TimeInterval$
A51: $AbstractRegion \sqsubseteq Region$

Figure 1: Graph of axioms

A given axiom may match multiple rules and therefore have multiple possible realisations. For instance, the axiom:

$$Student \equiv Person \sqcap \exists Supervisor.Academic$$

would be mapped to “A student is a person with at least one academic supervisor”, which exploits knowledge of the lexical categories of the names used, but another possibility would be something like “Something in class student is something in class person with at least one value for the role supervisor which is something in class academic” (this might have been the only possibility if the class names had been arbitrary identifiers such as “Class1” and “Class2”.).

Where a logical formula has multiple realisations, a measure of linguistic complexity of the results can be used to select a preferred one. Currently we measure linguistic complexity as the number of words in the English string generated. Better measures will take into account the shape of the parse tree. Notice that linguistic complexity does not directly mirror the complexity of the formula, but depends on ρ and whatever linguistic resources underlie it. Although more complex formulae tend to yield more complex linguistic output, linguistic complexity is also affected by:

- The extent to which special-case shorter rules match some of its subexpressions
- The extent to which class and role names can be interpreted as English words of relevant classes
- Whether a recursive linguistic structure uses left, right or centre embedding [Miller and Isard, 1964]

The linguistic complexity of a formula is obtained by taking the linguistic complexity of the realisation that is least complex. Again, although there is a correlation with the complexity of the formula, the relevant complexity for deciding, for instance, whether a formula can be presented in a single sentence, is a linguistic one which needs to take ρ into account.

[1] Note that our initial approach is to see how much can be achieved with no restrictions on the ontology (as long as it is expressed in legal OWL DL) and only generic linguistic resources (such as WordNet [Miller, 1995]). This is partly because there is a need to present parts of current ontologies, which often come with no consistent commenting or linguistic annotations, and partly so that we can then make informed recommendations about what kinds of extra annotations would be valuable in the ontologies of the future. Also, note that the term “realisation” will be taken here to include elements of “microplanning” which, for instance, introduces appropriate pronominalisation.

[2] We cannot guarantee that the ontology writer will use such mnemonic names (if not, then generation will have to use the less optimised templates), but we should exploit these cases when they arise (and our investigations have shown that they are extremely common in human-written ontologies).

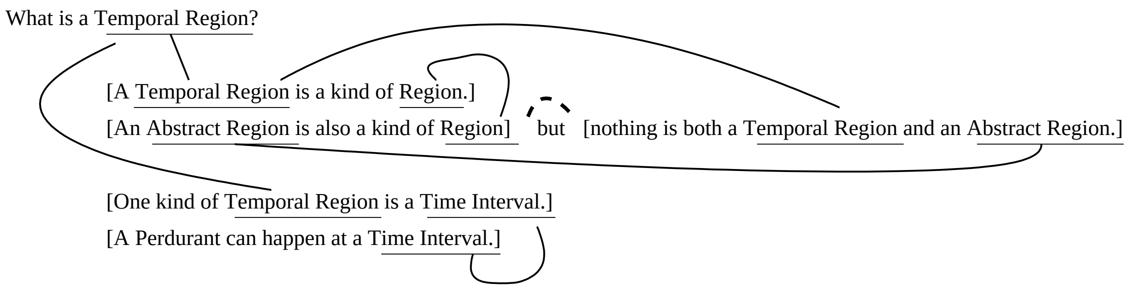

What is a Temporal Region?
[A Temporal Region is a kind of Region.]
[An Abstract Region is also a kind of Region]
but
[nothing is both a Temporal Region and an Abstract Region.]
[One kind of Temporal Region is a Time Interval.]
[A Perdurant can happen at a Time Interval.]

Figure 2: Example text with coherence relations shown

3 Selecting Material

The designer of an ontology has chosen one of many possible logically equivalent ways to axiomatise their domain, and this is important information. Therefore our initial approach worked from the axioms themselves without manipulating them in any way.

We basically followed the same procedure for content determination as in the ILEX system [O'Donnell *et al.,* 2001]. Thus the axioms can be seen as forming a graph, where each axiom is connected to the concepts it mentions (and where there may also be other links for relations between axioms) – see Figure 1. In this graph, routes between axioms correspond to different possible transitions in a coherent text – a text proceeds from one sentence to another by exploiting shared entities or by virtue of a rhetorical relation between the sentences[3].

A possible hand-generated text from the above axioms, showing the coherence relations which hold by virtue of shared entities or a rhetorical relation (the latter shown in dashes) is shown in Figure 2.

Assuming for the moment that a user has asked the question *What is X?*, where X is some class used in the ontology, selecting the axioms to express in the answer involves a best-first search for axioms, starting at the entity X. Each axiom is evaluated according to:

- how close it is (in terms of edges of the graph) to the concept X, and
- how intrinsically interesting, important and understandable it is.
- how few times is has already been presented

Following ILEX, these three measures are multiplied together and, for a text of length n, the n facts with the highest measures are selected for inclusion. The first component of the measure ensures that the retrieved axioms are relevant to the question to be answered. In terms of this, the best axioms to use are ones directly involving the class X. On the other hand, axioms that are only indirectly involved with X can be selected if they score well according to the second component (or if there are not enough closer axioms). The fact that there is a path between X and each chosen axiom ensures that there is a way of linking the two in a coherent text, by progressively moving the focus entity of the text to new entities in the axioms already expressed or through expressing rhetorical relations.

The second component of the evaluation score for axioms can be used to make the system sensitive to the user, for instance by preferring axioms that involve concepts known to the user or axioms that have not previously been told to them. We have not yet exploited this feature. The third component penalises axioms that have already been presented.

4 Natural Language Directed Inference

The content determination approach just described, which selects from among the provided axioms, suffers from a number of deficiencies:

Over-complex sentences: The axioms may not package the available information appropriately for natural language sentences. On the one hand, an axiom may be too complex to express in a single sentence (as determined by applying ρ and measuring the linguistic complexity). In this case, it might be appropriate to present a "weaker" (axiom). For instance, instead of expressing $X \equiv Y \sqcup Z \sqcup \ldots$ one might express $Y \sqsubseteq X$ (if it mentions the entities needed for coherence with the rest of the text).

Repetitive sentences: On the other hand, the axioms may give rise to sentences that are short and repetitive. Thus, rather than using three sentences to express:

$$Student \sqsubseteq Person$$
$$Student \sqsubseteq UnEmployed$$
$$Student \sqsubseteq \exists Supervisor.Academic$$

one could combine them all into a formula realised as "a student is an unemployed person with at least one academic supervisor". In NLG, the process of building such complex sentences is known as "aggregation" [Shaw, 1995]. This kind of aggregation could be implemented by combining the axioms together before realisation is performed, but success can only be measured by looking at the *linguistic* complexity of the result.

[3] The ILEX model makes use of the idea that there may be rhetorical relations, such as *concession* or *background*, between facts which could potentially be expressed in a text. It is however not immediately clear how they arise in our context. It seems that it may be plausible to say, for instance, "Although students are people and lecturers are people, lecturers and students are disjoint", but the general principles for this need to be worked out.

Inappropriate focus: An axiom may be expressed in a way that, when realised, places inappropriate emphasis on entities. For instance, an axiom $X \sqsubseteq Y$ could be realised by "An X is a kind of Y", whereas the equivalent $Y \sqsupseteq X$ could be realised by "Y's include X's". The latter would be much better than the former at a point in a text that is discussing the properties of Y. The above example of "weakening" also has the effect of changing the likely subject of the sentence produced. Sometimes the text will be better if one can switch around the material in an axiom to emphasise different material.

Misleading partial information: It may be better to present some of the consequences of an axiom, given the rest of the theory, rather than the axiom itself. For instance, instead of presenting

$$Student \sqsubseteq \exists supervisor.Academic$$

in an ontology which also has the axiom $functional(supervisor)$, it would be more informative to present the consequence

$$Student \sqsubseteq\ = 1\ supervisor.Academic$$

Indeed, with number restrictions a reader can draw false implicatures (in the sense of [Grice, 1975]) if only partial information is presented. In this case, a scalar implicature [Levinson, 1983] is involved. A reader, on being told that "a student has at least one academic supervisor", will naturally assume that they could have more than one, or that they could have other supervisors belonging to other classes. Similarly, on being told "a supervisor of a student is always an academic", one will assume that there can be more than one supervisor (otherwise the text would have said "*the* supervisor ..."). Some of the principles at work here may be similar to those encountered in cooperative question answering [Gaasterland *et al.*, 1992].

The only way to overcome these limitations is to enable content determination to select material in more ways than just choosing an axiom. It must always choose to express something that is true, given the logical theory, and content determination will therefore be a form of *inference*. In general, in fact, we could consider using any logical consequence of the axioms. However, not all logical consequences are equally good. The formulae that are presented should:

1. **Soundness:** follow from the original logical theory (set of axioms)
2. **Relevance:** contribute information relevant to the goal of the text. For instance, if the goal is to answer the question "*what is concept X?*" then the formulae should be about X or other concepts which shed light on X.
3. **Conservatism:** be not very different from the original axioms (and so capture some of the intent behind those axioms)
4. **Complexity:** have appropriate linguistic complexity (section 2)
5. **Coherence:** satisfy linguistic coherence constraints (i.e. be linked to other selected material by the kinds of relations discussed in section 3).
6. **Novelty:** not have already been expressed (and not be tautologies). There is no point in weakening axioms to the point that nothing new is expressed, or in presenting the same material many times.
7. **Fullness:** be complete, to the extent that they don't support false implicatures
8. **User-orientation:** be in accord with user model preferences (as in section 3)

We call the kind of inference required to find such logical consequences *natural language directed inference* (NLDI). It is a kind of forwards inference with very specific goals, which arise from its use for natural language generation.

Although we have motivated NLDI through our own particular content determination problem, this may be a useful way to view content determination in general, as long as the starting point can be viewed as some kind of logical theory, Θ, there is an available realisation relation ρ and an evaluation function $eval$ for linguistic outputs, which takes into account the above desiderata. In this case, content determination can be viewed as the problem of determining

$$argmax_{(\theta\ such\ that\ \Theta \models \theta)}\ max\{eval(t) | t \in \rho(\theta)\}$$

The process of enumerating promising consequences of Θ for this optimisation is certainly a form of logical inference. But its goal is unlike standard goals of automated reasoning and is shaped by the idiosyncracies of the requirements for natural language output. There is an interesting parallel here with the work of [Sripada *et al.*, 2003]. Sripada *et al* found that, for generating natural language summaries of time series data, standard data analysis algorithms such as segmentation had to be modified. They characterised the extra requirements that forced these modifications in terms of the Gricean maxims of cooperative communication [Grice, 1975]. Our 8 desiderata above could also be thought of as cases of the Gricean maxims.

5 Techniques for NLDI

Unfortunately, standard refutation-based approaches to inference rely on having a precisely specified inference goal, whose negation is incompatible with the axioms. For DLs, the standard tableaux methods [Horrocks, 1998] have similar properties. NLDI does not have an inference goal that can be expressed in structural terms, so even approaches to "matching" cannot straightforwardly be used to derive linguistically appropriate results. NLDI is more akin to other "non-standard" types of inference, perhaps to approximation [Brandt *et al.*, 2002], though again the target logical language is without a simple formal characterisation. Perhaps the closest approach we are aware of is meta-level control of inference, where factors outside of the logic (e.g. other kinds of descriptions of the shapes of logical formulae) are used to guide inference [Bundy and Welham, 1981].

One advantage of NLDI is that it does not have to be a complete inference procedure, though in general the more logical consequences of the original axioms it can find, the more possible texts will be considered and the higher the quality of the one chosen.

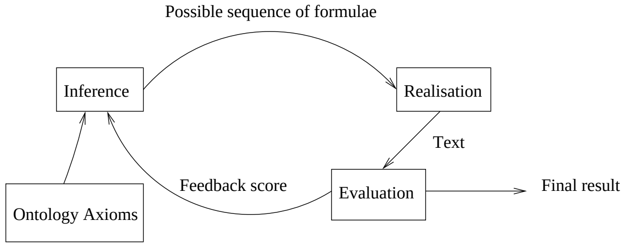


Figure 3: Overgeneration Architecture

The approach to NLDI we are currently working on is inspired by the idea of "overgeneration" approaches to NLG, as used, for instance, by those using statistical models [Langkilde and Knight, 1998] and instance-based search [Varges and Mellish, 2001]. In this approach, instead of attempting to intelligently order the relevant choices to come up with an optimal text, an NLG system consciously enumerates a large number of possible texts (in a cheap way) and then chooses between them using a linguistically-aware evaluation function of some kind (the $eval$ of NLDI). Our approach differs from these others, however, in that, whereas the other systems implement overgeneration of *surface forms*, we consider overgeneration of possible *content*.

Figure 3 shows the architecture of our system under development. The simple inference system implements a beam search among possible sets of content for generating texts, where each state in the search space is a sequence of formulae. In logical terms, each sequence represents a conjunction that follows from the input axioms. The resulting text for any such sequence (i.e. the result of applying ρ) will be the result of realising the elements of the sequence, in order, as the sentences of the text.

At each point in the search, the current state can give rise to new states in two possible ways:

1. One of the original axioms is added to the end of the sequence.
2. The final formula of the sequence is replaced by a formula inferred from it (given the whole axiom set) by one inference step

The inference steps represent simple ways of modifying a formula to something close to it which follows from the complete set of axioms and which may yield a more appropriate realisation. We have currently implemented a small number of relevant steps, including steps for aggregation, disaggregation and elimination of disjunctions.

Whenever a new state in the search space is generated, it is sent to the realisation component (which implements ρ) and from there through an evaluation function (which implements $eval$). The evaluation function takes into account the average deviation of the sentence lengths (in words) from an "ideal" sentence length and some other heuristics (see below). This is used as feedback to drive the search of the inference component in a best-first manner. The search terminates when the best scoring state is one element longer that the desired number of sentences for the text, at which point its sequence of formulae, apart from the last one, is returned. The exploration to a length longer than the desired one ensures that other states shorter than or equal to the desired length have a chance to be explored.

Our approach makes initial attempts to address the desiderata of NLDI by constraining the search in the following ways:

1. Soundness: All new formulae are derived by sound rules of inference from the existing axioms and so are true.

2. Relevance: Only axioms which might affect the interpretation of the class asked about are ever considered (the rest are discarded at the start of the process). For the purposes of this, we use the conservative relevant-only translation of [Tsarkov *et al.*, 2004] to discard axioms that cannot be relevant to the question.

3. Conservatism: Inferred formulae are based on individual axioms, and shorter inferences are enumerated before longer ones.

4. Complexity: The complexity of the best realisations is used to order the search candidates. Candidates which are inappropriate for realisation do not match the realisation rules and so are not considered.

5. Coherence: When a new axiom is added to a sequence, it is constrained in its realisation to have a subject which is a class mentioned in the previous element of the sequence. The subject of the first element of the sequence must be the class which is the subject of the original question. Also the evaluation function has a preference for the first sentence with a given class as subject to be an "is a" type sentence.

6. Novelty: In order to prevent information being presented more than once, only one logical consequence of any given axiom is ever included in a sequence. This is implemented via a simple way of tracking the axioms that have contributed to each formula. This makes the assumption that the original axioms are logically independent.

7. Fullness: Formulae are closed with respect to cardinality information before being added to the lists.

8. User-orientation: We don't currently take this into account, but intend to reward formulae that contain class and role names already familiar to the user (e.g. used in answers to previous questions, or appearing earlier in the answer to the current question).

All of these are relatively crude measures, which nevertheless give some appropriate direction to the process.

This system has been implemented and tested informally on examples from three different ontologies. For example, in creating a 3-sentence text to answer "What is an Electrode?" using a fuel cell ontology with 133 axioms, the relevance filter first of all reduces the set of axioms to 31, which include:

(1) $Electrode \sqsubseteq Actuality$
(2) $Electrode \sqsubseteq \exists contains.Catalyst$
(3) $Electrode \sqsubseteq (\exists contains.Support \sqcap \leq 1\ contains.\top)$
(4) $domain(contains, FuelCell \sqcup MEA \sqcup Electrode \sqcup Catalyst)$

as well as other axioms such as $Catalyst \sqsubseteq \exists contains.ActiveMetal$. If these 4 axioms were selected unchanged and realised in this order (which by chance happens to be quite a good order), then the following text would result:

> An Electrode is a kind of Actuality. An Electrode always contains something which is a Catalyst. An Electrode always contains something which is a Support and always contains at most 1 thing. Only something which is a FuelCell, a MEA, an Electrode or a Catalyst contains something.

Instead of this, our simple implementation of NLDI proceeds as follows. The initial states are those axioms which when realised will have $Electrode$ in subject position, with a preference for those that will be realised as "an Electrode is a ...". Thus the state consisting of the one element sequence:

$$Electrode \sqsubseteq Actuality$$

will be a favourite. This state can be developed in several ways. For instance, another axiom could be aggregated with this one (to give a sentence of the form "an Electrode is an actuality which ..."). Another possibility is for this axiom to be accepted in this form and for another axiom to be added to the end of the sequence. This second possibility generates the following state, among others:

$$Electrode \sqsubseteq Actuality$$
$$Electrode \sqsubseteq = 1\, contains.Catalyst$$

(notice how more precise cardinality information has been attached to axiom (2)). This state can be further developed by adding a further axiom, or by applying an inference rule to the last added formula. In this case, aggregation with axiom (3) is possible, yielding:

$$Electrode \sqsubseteq Actuality$$
$$Electrode \sqsubseteq = 1\, contains.(Catalyst \sqcap Support)$$

This state is further developed by adding new axioms to the end, and so on. The final sequence of formulae selected is:

$$Electrode \sqsubseteq Actuality$$
$$Electrode \sqsubseteq = 1\, contains.(Catalyst \sqcap Support)$$
$$domain(contains,$$
$$FuelCell \sqcup MEA \sqcup Electrode \sqcup Catalyst)$$

This is realised by the following short text:

> An Electrode is a kind of Actuality. An Electrode contains exactly one thing, which must be a Catalyst and a Support. Only something which is a FuelCell, a MEA, an Electrode or a Catalyst contains something.

(This realisation relies on part-of-speech information which can be obtained automatically from WordNet, apart from the term "MEA").

6 Discussion

Although NLG lacks a general account of content determination, one area of content determination that *has* been well formalised is the problem of generating referring expressions. Here the task is to find a *distinguishing description* of an entity that it is true of the entity but not of any of the "distractors" in some current context. Recent work has formalised NLG algorithms for referring expression generation in terms of algorithms for finding an appropriate subgraph of a graph representing the domain knowledge [Krahmer *et al.*, 2003]. Given that the graphs involved are very similar to Conceptual Graphs [Sowa, 1984] and that the projection relation between Conceptual Graphs (an extended notion of subgraph) is a kind of inference, it follows that these referring expression algorithms can also be viewed as performing inference. As work considers an increasing range of referring expressions (e.g. using relations, logical connectives, plurals and even quantifiers), the complexity of the inference required is forcing researchers increasingly to depart from the original graph matching approach. We believe that it may well prove productive to view this as a case of NLDI, especially as (in spite of the assumptions of most current work) logical complexity and linguistic complexity are not always the same.

There are many issues to be addressed in the development of a convincing approach to NLDI. For instance, it is necessary to determine what kinds of inference steps are relevant to the optimisation of linguistic properties. In our system, we would certainly like to introduce unfolding operations to steer the system towards using concepts that the reader is familiar with. In addition, ideas from linear logic [Girard, 1987] may be relevant to avoiding duplication in the information conveyed. Finally, there are real questions about the ideal architecture of an NLDI system. If the $eval$ function or ρ is expensive, then it may be necessary to interleave the evaluation and the inference steps more than we have done, to the extent that inference is directly aimed at achieving linguistic effects.

Acknowledgments

This work is supported by EPSRC research grant GR/S62932. Many thanks to Ian Horrocks, Alan Rector and members of the Aberdeen NLG group for useful discussions.

References

[Bontcheva and Wilks, 2004] K. Bontcheva and Y. Wilks. Automatic report generation from ontologies: the miakt approach. In *Ninth International Conference on Applications of Natural Language to Information Systems (NLDB'2004)*, Manchester, UK, August 2004.

[Brandt *et al.*, 2002] S. Brandt, R. Küsters, and A.-Y. Turhan. Approximation and difference in description logics. In D. Fensel, D. McGuinness, and M.-A. Williams, editors, *Procs of KR-02*. Morgan Kaufmann Publishers, 2002.

[Bundy and Welham, 1981] Alan Bundy and Bob Welham. Using meta-level inference for selective application of multiple rewrite rule sets in algebraic manipulation. *Artificial Intelligence*, 16(2):111–224, May 1981.

[Gaasterland *et al.*, 1992] T Gaasterland, P Godfrey, and J Minker. An overview of cooperative answering. *Intelligent Information Systems*, 1(2):123–157, 1992.

[Girard, 1987] J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[Grice, 1975] H. P. Grice. Logic and conversation. In P. Cole and J. Morgan, editors, *Syntax and Semantics: Vol 3, Speech Acts*. Academic Press, 1975.

[Horrocks, 1998] Ian Horrocks. Using an expressive description logic: Fact or fiction? In *Proceedings of the 6th International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*, pages 636–647, 1998.

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

[Langkilde and Knight, 1998] I. Langkilde and K. Knight. Generation that exploits corpus-based statistical knowledge. In *Proc. of the Conference of the Association for Computational Linguistics (COLING/ACL)*, 1998.

[Levinson, 1983] S. C. Levinson. *Pragmatics*. Cambridge University Press, 1983.

[McGuinness and van Harmelen, 2004] D. L. McGuinness and F. van Harmelen. Owl web ontology language overview. `http://www.w3.org/TR/owl-features/`, 2004.

[Mellish *et al.*, 1998] C. Mellish, M. O'Donnell, J. Oberlander, and A. Knott. An architecture for opportunistic text generation. In *Proceedings of the ninth international workshop on natural language generation*, pages 28–37. Association for Computational Linguistics, 1998.

[Mellish *et al.*, 2004] C. Mellish, M. Reape, D. Scott, L. Cahill, R. Evans, and D. Paiva. A reference architecture for generation systems. *Natural Language Engineering*, 10(3/4):227–260, 2004.

[Meteer, 1992] Marie Meteer. *Expressibility and the Problem of Efficient Text Planning*. Pinter publishers, London, 1992.

[Miller and Isard, 1964] G. Miller and S. Isard. Free recall of self embedded english sentences. *Information and Control*, 7:292–303, 1964.

[Miller, 1995] G. Miller. Wordnet: A lexical database for english. *CACM*, 38(11):39–41, 1995.

[Moore, 1994] Johanna Moore. *Participating in Explanatory Dialogues*. MIT Press, 1994.

[O'Donnell *et al.*, 2001] M. O'Donnell, A. Knott, C. Mellish, and J. Oberlander. *ILEX*: The architecture of a dynamic hypertext generation system. *Natural Language Engineering*, 7:225–250, 2001.

[Shaw, 1995] James Shaw. Conciseness through aggregation in text generation. In *Procs of the 33rd Annual Meeting of the Association for Computational Linguistics*, MIT, 1995.

[Shieber, 1993] Stuart Shieber. The problem of logical-form equivalence. *Computational Linguistics*, 19(1):179–190, March 1993.

[Sowa, 1984] J. F. Sowa. *Conceptual Structures - Information Processing in Mind and Machine*. Addison-Wesley, 1984.

[Sripada *et al.*, 2003] S. Sripada, E. Reiter, J. Hunter, and J. Yu. Generating english summaries of time series data using the gricean maxims. In *Proceedings of the Ninth ACM SIGMOD International Conference on Knowledge Discovery and Data Mining (KDD-2003)*, pages 187–196, 2003.

[Tsarkov *et al.*, 2004] Dmitry Tsarkov, Alexandre Riazanov, Sean Bechhofer, and Ian Horrocks. Using vampire to reason with owl. In Sheila A. McIlraith, Dimitris Plexousakis, and Frank van Harmelen, editors, *Procs of the 2004 International Semantic Web Conference (ISWC 2004)*, pages 471–485. Springer LNCS 3298, 2004.

[Varges and Mellish, 2001] S. Varges and C. Mellish. Instance-based natural language generation. In *Procs of NAACL-01*. Carnegie Mellon University, 2001.

[Wagner *et al.*, 1999] J. Wagner, J. Rogers, R. Baud, and J-R. Scherrer. Natural language generation of surgical procedures. *Medical Informatics*, 53:175–192, 1999.

[Wilcock, 2003] G. Wilcock. Talking owls: Towards an ontology verbalizer. In *Human Language Technology for the Semantic Web and Web Services, ISWC-2003*, pages 109–112, Sanibel Island, Florida, 2003.