Web Intelligence and Agent Systems: An International Journal 5 (2007) 1–5
IOS Press

Gen-Meta: Generating metaphors by combining AI and corpus-based modeling

Andrew Gargett [a,*] and John Barnden [b]

[a] *School of Computer Science, University of Birmingham, Edgbaston, Birmingham, B15 2TT, United Kingdom*
E-mail: a.d.gargett@cs.bham.ac.uk

[b] *School of Computer Science, University of Birmingham, Edgbaston, Birmingham, B15 2TT, United Kingdom*
E-mail: j.a.barnden@cs.bham.ac.uk

Abstract. Metaphor is important in all sorts of mundane discourse: ordinary conversation, news articles, popular novels, advertisements, etc. Issues of prime human interest – such as relationships, money, disease, states of mind, passage of time – are often most economically and understandably conveyed through metaphor. This ubiquity of metaphor presents a challenge to how Artificial Intelligence (AI) systems not only understand inter-human discourse (e.g. newspaper articles), but also produce more natural-seeming language; however, most AI research on metaphor has been about its understanding rather than its generation. To redress the balance, we directly tackle the role of AI systems in communication, uniquely combining this with corpus-based results to guide output toward more natural forms of expression.

Keywords: Artificial intelligence, Cognitive science, Natural language processing, Interactive system, Corpus Linguistics

1. Introduction

Working out why a speaker might choose to use metaphor is very much an open question. By way of attempting to answer the more tractable question of why, after having decided to express things metaphorically, a speaker may choose one metaphorical expression over another, we have formulated a way of meeting the challenge of generating metaphor. We describe in this paper an approach to metaphor generation which uniquely combines reasoning with data-oriented techniques, potentially accounting not only for more conventional forms of metaphorical expression, but also novel extensions to established metaphor. Our approach, which we have dubbed "Gen-Meta",[1] provides a Natural Language Generation (NLG) front-end for a state-of-the-art metaphor-based processing framework, ATT-Meta [5]. Based on patterns of metaphorical expressions, mined via various methods for discovering metaphor in natural language, our system links an AI-derived conceptual level to a corpus-derived linguistic level, thereby generating an appropriate expression for the target metaphorical meaning.

While still being prototyped, the system aims to coordinate in modular fashion the interaction between three existing frameworks: ATT-Meta, Embodied Construction Grammar [16], and Dynamic Syntax [29]. This paper reports on the development of this approach, as well as some initial findings.

2. Generating metaphor

2.1. Overview of Natural Language Generation (NLG)

Producing natural language utterances involves numerous choices about *what* to say and *how* to say it;

*Corresponding author. E-mail: a.d.gargett@cs.bham.ac.uk. Phone: +44-121-414-6749. Fax: +44-121-414-4281.

[1] http://www.cs.bham.ac.uk/ gargetad/genmeta-index.html

1570-1263/07/$17.00 © 2007 – IOS Press and the authors. All rights reserved

how to best make these choices are central problems in Natural Language Generation (NLG), the study of the use of computational techniques for adequately generating strings of natural language. Such choices cover everything from deciding the basic content of the utterance (*what*), through to determining how to resolve forms of reference, planning discourse structure, and realising appropriate words and their combinations [11]. Regarding approaches to modeling such decision-making in NLG, there seem to three broad classes:

1. Template-based: generating via predefined slots of a template
2. Pipeline-based: stepping through decisions about what to say and how to say it, like a sort of production line
3. Learning-based: adapting via a form of learning, to particular domains and/or users [25]

The first class is the most common, while the second and third are closest to our own, combining "knowledge intensive" approaches to metaphorical extension through inferential processing, with more data-oriented approaches crucial for modeling the wide variety of forms possible for expressing oneself metaphorically. While much (if not all) NLG takes actual usage into account, we directly incorporate patterns of metaphorical expression found in corpora, to produce texts more directly reflecting language use.[2]

2.2. *Key issues in generating metaphor*

Producing expressions that are in some sense "more natural" is a key aim in NLG, so that a phenomenon as ubiquitous in everyday human communication as metaphor (e.g. [23], [10]) should be a priority within NLG, one would think. Yet, while there is a recognisable body of research on the natural language understanding (NLU) of metaphor, much less research has been devoted to generating metaphor [19]. Both NLU and NLG face many of the same issues when modeling more general cognitive phenomena such as metaphor, which apparently require solutions to substantial parts of core artificial intelligence. While much NLG research has assumed content to be given, enabling a focus on *how* to realise such content in actual linguistics strings, generating metaphor is very much about modeling content, requiring new ways of thinking and new techniques, or at least fresh ways of using already established techniques.

Opting to metaphorically express an idea implies strategically choosing this form of expression over another, such as in emotionally charged encounters [10]. No current NLG system can generate metaphor in a way that is contextually appropriate, as humans do all the time when communicating with one another, yet there has been a variety of previous attempts at generating metaphor, some of which we consider below.

2.3. *Inferential approaches to modeling metaphor*

Past approaches to metaphor generation based on rule- or constraint-based methods include [26], [21], [32] and [19]. We have chosen to compare two exemplary approaches.

2.3.1. *MIDAS*

The "computational theory of metaphor" proposed by [26] yielded the MIDAS system, having the capacity to both understand and generate metaphors in the narrow domain of the UNIX operating system, for example:

(1) How can I *get into* mail?
(2) How can I *get out of* emacs?
(3) How can I *kill* a file?

The italicised items are metaphorical, since a direct reading of verbs is non-sensical in these contexts (e.g. killing a file here cannot mean, directly, ending the life of something that is alive, but it can mean, less directly, ending a computer process, and even deleting some item of information stored on a computer). For [26], many such metaphors are largely conventional (see [23]), reflecting larger conceptual classes of which they are members (other examples being ARGUMENT AS WAR, TIME AS MONEY). MIDAS stores such conventional metaphors in its lexicon, this being an instance of a knowledge-rich approach to metaphor processing, whereby understanding a particular (conventional) metaphor is largely a matter of being able to access the entry for that metaphor.

An important capability of MIDAS is its ability to go beyond its current store of conceptual mappings. [26] points out that MIDAS deals with unfamiliar metaphors "by extending an existing metaphor in

[2] We consider our approach to address the so-called "knowledge bottleneck" [24], by tackling the immense amount of (lexical, morphological, syntactical, etc) knowledge required to generate natural language. Of course, the challenge of building the resources required for work on large-scale corpora is not without problems (see e.g. [3]).

a systematic fashion to cover a new use." The chief mechanism whereby MIDAS achieves this is through one of three forms of extension: *similarity extension*, *core extension*, and *combined extension*. These extension types are defined and illustrated in Table (1). [26] concedes that under this approach, if a metaphor is truly novel in the sense of being unrelated to the current store of knowledge about metaphor, then it is unlikely to be understood by the system.

[26] also presents an implementation of such metaphorical extension, demonstrating the innovative nature of his approach. MIDAS is an important predecessor to our own approach to generating metaphor.

[15] notes that while MIDAS is apparently overspecialised to the domain, incorporating conventional metaphors like IS AT VARIABLE VALUE, which may have been "added to MIDAS especially for interpreting particular sentences", the coverage of MIDAS is certainly impressive. Since MIDAS, there have been few knowledge-rich approaches with a substantially greater coverage. Here, as elsewhere in NLG, how to model content adequately has been the chief obstacle to progress.

2.3.2. ATT-Meta

A key component of our approach to metaphor is the ATT-Meta system, a state-of-the-art AI system for modeling metaphor as reasoning-by-simulation [5], whereby those aspects of a metaphorical expression, like *How do I get out of emacs?*, which are clearly not about reality, on a par with *How do I get out of this house?*, are dealt with in a distinct mental space, a so-called metaphorical pretence cocoon, wherein reasoning about such propositions and inferences can be kept separate from propositions and reasoning about reality.

While ATT-Meta has until now been used for metaphor understanding, it turns out to be fairly straightforward to extend it to generation, due to a novel feature of the system, namely its ability to transfer information from target-to-source, as well as in the more usual source-to-target direction. The reversed transfer is held to be crucial for the understanding of some metaphor, but can be adapted also for generation. ATT-Meta is set apart from other approaches to metaphor in AI not only by its use of reverse transfer and pretence cocoons but also by allowing metaphorical transfer steps to be freely intertwined with general reasoning about the target and source domains, where that reasoning is itself of indeterminate extent. This allows great generality in the metaphorical language it can address, and great context-sensitivity of interpretation.

While ATT-Meta's reverse use of mappings can be readily deployed as a part of the process of generating metaphorical utterances, we need some way of causing a reverse use to happen, bearing in mind that ATT-Meta works entirely by backward-chaining reasoning, or goal-directed reasoning, a form of reasoning commonly used in rule-based systems.So we need either to add a forward-chaining capability to ATT-Meta (so that, given a reality-space representation, reasoning would step forwards into the pretence space across a mapping), or to emulate such forward chaining by construcing a certain type of rule of the following intuitive form (where "can-state(Y)" means that the system can create linguistic output that expresses Y):

(R1) IF reality situation X corresponds to pretence situation Y, and Y holds in the pretence THEN can-state(Y).

where X and Y are variables. Here we are helped by a distinctive feature of ATT-Meta mappings, in that they have the form:

(R2) IF guard-condition G holds THEN real-U corresponds to pretend-V.

(R3) IF X is a person and D is a disease and in the pretence D is a physical object, THEN **ep**(being_infected, X, D) in reality corresponds to **ep**(physically_possessing, X, D) in the pretence.

Here and throughout, **ep** is shorthand for "the episode", and refers to an instance of some general event type, such as "being_infected" in **(R3)**. Thus, rule **(R2)**, and its instantiation in **(R3)**, would only pick up those mappings whose guard conditions are satisfied. Then crucial to understanding the claim that Bill *has* a cold, is presuming a cold to be a physical, and hence possessable, object, permitting only mappings whose guards (antecedents in these conditional rule forms) are satisfied by that presumption.

Consider the following example utterances expressing the metaphorical notion of a cold as a physical thing (including representations of these in ATT-Meta terms):

(4) *Bill has a cold* $\rightarrow$ **ep**(phys_possessing, bill, cold) [in the pretence]
(5) *Bill gave Bob a cold* $\rightarrow$ **ep**(phys_transferring, bill, bob, cold) [in the pretence]

Table 1

The three forms of metaphorical extension employed by MIDAS (from [26]).

Extension type	Example
Similarity: The newly encountered metaphor is understood in light of a known metaphor	(e.g.1) The unfamiliar *Kill a process* is similar to *Kill a conversation*
Core: The newly encountered metaphor is understood by virtue of a metaphor considered to be closely related	(e.g.2) *Get out of lisp* is unfamiliar yet closely related to *Get into lisp*
Combined: Employing both similarity **and** core extensions	(e.g.3) Newly encountered *Exit emacs* is both similar and closely related to the more familiar *Enter email*

First consider understanding of the sentence *Bill has a cold* assuming that *has* is interpreted as physical possession. Through use of, in part, **(R3)** to infer a correspondence concerning John's cold, and then using this correspondence in an act of source-to-target transfer, ATT-Meta can create the following from the RHS of (4):

(6) **ep**(being_infected, bill, cold) [in reality]

For generation, suppose we start with (6), and suppose that in the discourse John's cold has already been metaphorically regarded as a physical object, or (more likely for this example) it is a standard default assumption that a cold is metaphorically regarded as a physical object (i.e., it is a physical object in the pretence). Then only those mapping rules such as **(R3)** whose guards are satisfied by that condition will be enabled. So, a specific instance of the correspondence in **(R3)** comes to hold, namely the instance with X=John and D=cold, rather than having such a correspondence hold of anyone and of any disease. So, John's having his cold is deemed to correspond to John's physically-possessing his cold (as in (4)). Thus, rule **(R3)**, by the very fact of picking up on such specific mapping instances, will already at least partially instantiate Y to that specific situation.

By way of illustrating further the approach taken by ATT-Meta to modeling relations between reality and pretence space, it should also be noted that, given that we are attempting to model talk about diseases, ATT-Meta has provisions for treating some physical objects as copiable, so that transferring them doesn't cause the original owner to lose the object. Part of ATT-Meta's built-in knowledge of the metaphorical view of diseases as physical objects is a default *ancillary assumption* that when a disease is being regarded as a physical object, it is in fact a copiable object.

Finally, a note of clarification about our examples (the antecedents in(4) and (5)) may be necessary – these examples are being used purely for a simple illustration of how ATT-Meta works, and under the assumption, for the sake of presentation, that using words like *have* and *give* about a disease is not highly conventional phraseology (although, arguably it is in fact conventional to a significant degree). Importantly, the type of example that ATT-Meta is designed for is typified by the (attested) example in (7):

(7) I don't think strings are attached. If they're there they're nylon ones. I don't see them.

Here, a conventional metaphor of abstract constraints as strings is elaborated to bring in the unconventional issue of the physical composition of those strings, requiring reasoning about the properties of nylon.

2.4. *Data-driven approaches to modeling metaphor*

Mounting evidence suggests people frequently employ formulaic language to express figurative meanings such as metaphor (e.g. [14], [20]). The approaches of [13] and [10] are central to how we ourselves approach this. [13] prefers a corpus-driven approach to modeling metaphor, aiming to determine taxonomies directly from the corpora concerned, although she points out that many corpus-driven studies of metaphor, tend to start "by necessity... with some sort of working hypothesis, but this is explored and tested through the data rather than being preimposed on them." [10] presents evidence of metaphor *tuning* during reconciliation talks within the context of acts of terrorism, in particular the way in which someone can increase the impact of their contribution by employing metaphor to describe the effect on their lives of another's actions. Such evidence has inspired our own use of corpus studies to guide generation toward conventionalised forms of expressing metaphors (for further illustrative approaches, see also [1], [31]). For example, one interesting data-intensive method to generating metaphor is [31], which involves mining the world-wide web in order to process and generate "apt" metaphors, finding metaphorical expressions based on the grammatical markedness of relatively simple similes, schematically ***T[enor] is as P[roperty] as [a] V[ehicle]***; from these can be extracted information about the target and sources involved (e.g. that **P** is a salient aspect of **T**, is shared by both **T** and **V**, etc). Mining the web like this provides their system, Sardonicus, with a set of cases from which it can generate metaphorical expressions which are deemed apt, perhaps due to their relatively high frequency of occurrence, demonstrating the benefits of employing a data-intensive approach to generating metaphor that is able to exploit large amounts of data. Now, a consequence of their approach is that finding a vehicle for a term like graceful is potentially an open-ended task (e.g. *as a swan?*, *as an elephant?*), exploding the search space for choosing the most apt metaphor. Yet, with the help of a user, the search procedure is directed toward a goal, whereby the user specifies a tenor as well as a property to be focused on – e.g. for tenor *Paris Hilton* referring to a recent celebrity, the user can direct Sardonicus to generate apt metaphors focusing on the property *skinny*, with Sardonicus evaluating a range of possible vehicle nouns (e.g. *twig*, *pole*, *rake*, *cadaver*). [31] report possible metaphors that are returned through this method include (presented in Sardonicus format, bracketted numbers are counts):

post(46), pole(42), stick(38), miser(34), stick insect(26)

In turn, these can be analysed with respect to their properties:

straight(387), skinny(369), thin(353), slim(204), stiff(20), scrawny(8)

It is important to note that while Sardonicus has no knowledge base for Paris Hilton, since it is basically a purely data-intensive approach, lacking more sophisticated reasoning capacities, [31] point out that Sardonicus can be supplied with additional resources to make up for this limitation (e.g. hypotheses derived from collocational analysis of large corpora, and so working out the meaning of some lexical item based on the company it keeps).

3. Gen-Meta: combining inferential and data-oriented approaches to modeling metaphor

We will present our approach by starting with a brief overview, and then go on to report on the progress we have made in prototyping a system, by presenting a case study, as well as an empirical investigation we have carried out. These results are presented in terms of metaphorical expressions in illness discourse, which is one of the two domains we are investigating (the other being political conflict). Finally, although the current system is at the development stage, and is not yet at the point of producing metaphors automatically, we present plans we have for evaluating the system's output using both subjective and objective measures of this output.

3.1. Overview

Gen-Meta attempts to explicitly combine inferential and data-oriented modeling of metaphor, by chaining together modules which, as separately and independently validated approaches to modeling language, bring advantages to the resulting system as a whole. We have discussed ATT-Meta above, and immediately below we discuss the remaining two modules.

Embodied Construction Grammar (ECG) ECG is a language understanding (but not generation) system having aspects highly congenial to metaphor, and of interdisciplinary significance [17], [16].[3] ECG models the links between the conceptual level, represented as interconnected schemas, and the linguistic level, represented as interconnected constructions. Schemas consist of "roles" together with constraints on these roles. Recalling examples (4) to (6): there is a schema for the concept of somebody realising a **transferer role** TRANSFERRING something to somebody else realising a **transferee role**. ECG's schemas are strongly geared towards conceptual representations, and have broadly the same orientation as ATT-Meta's representations. Constructions will be familiar from work within Cognitive Linguistics [11], for example, the English verb *give* may employ a ditransitive construction having three constituents, which may be loosely referred to as **subject**, **direct object** and **indirect object** (see examples (4) to (6)) – ECG formally specifies the ordering constraints operating over such con-

[3] ECG has also recently been implemented [9].

stituents, as well as the linkage between the form and the meaning of the construction.[4] Being more flexible than traditional grammatical representations, constructions can better model the diversity and flexibility of expression types, from single words to multi-word expressions (MWEs). A particularly interesting example of a constructional combination of several lexical items, can be found in example (1), *How do I get into mail?*, similar to one discussed by [8]. The metaphorical expression of *into mail*, consists of constructions at both the lexical level (individual lexical items may be constructions in their own right), but also the supra-lexical level (the constructions expressed by the lexical items *into* and *mail*, being combined in the larger SPATIAL-PHRASE construction). Constructions also usefully represent conventional metaphorical form and meaning, with MWEs rather than individual words frequently having metaphorical meanings.[5]

Dynamic Syntax (DS) DS models syntax as a parsing-directed process of progressively developing linguistic content relative to context, with generation modeled as parsing-based so that DS is *bi-directional*. DS provides a fully incremental and context-dependent parsing model, with update modeled as transitions between successive parse states, and parsing proceeds essentially by incrementally enriching partial tree structures. Parsing can then be seen as the sequence of pairings of natural language strings of terms s with the logical formula LF representing the semantic structure of those terms, as in (8):

(8) $\{\langle s(i), \text{LF}_i\rangle, \langle s(i+1), \text{LF}_{i+1}\rangle, \ldots\}$

Thus, LF_i results from parsing $s(i)$. More generally, parse states are modeled in DS as triples $\langle PT, F_S, F_A\rangle$, of (i) (partial) tree structures PT, (ii) a function F_S mapping partial tree structures to items of the formal language, and (iii) a function F_A mapping actions (from sets of actions, A) for transition between pairs of partial trees. These trees represent semantic information, with syntax and lexical entries encoding instructions for building such trees, in a fully incremental as well as context-dependent manner. A key feature of the DS parser is its flexibility in being able to use not only (partial) trees from the current context, but also *reuse* (possibly partial) trees from some previous context (this capacity is an important aspect of the model of ellipsis in DS, e.g. [29]). Further, given the bi-directionality of the grammar, generation under this approach is just as dynamic, incremental, and context-dependent as parsing, and the resultant integration of generation and understanding makes DS well-suited to modeling dialogue [29]. In order to tie some of these points together, consider the following example:

(9) A: Bill gave Bob...
B: ...a cold, yes I know.

In this constructed dialogue, B completes A's utterance, and such completions are a common dialogue phenomena [28]. Note the similarity of (9) to elliptical phenomena, such as, "Bill gave Bob a cold, and Sue a bunch of roses," where the verb "give" is missing in the second clause but recoverable from the context that includes the initial clause. In the same way, in (9) B can treat their own understanding of A's utterance as the context against which they develop their own contribution, and the result of both A's and B's respective contributions is a complete utterance "Bill gave Bob a cold," split over two turns [29]. DS is able to model such phenomena quite naturally due to the tight integration of parsing and generation.

3.2. Assembling the system

Our project involves arranging these three standalone modules, ATT-Meta, ECG and DS, in a pipeline formation. Each component provides a necessary role in achieving a system with the capacity for generating metaphor. To this end, we develop a hybrid of DS and ECG, incorporating ECG-style form-meaning mapping, in particular between constructions and schemas, but extending ECG with parsing/generation techniques from DS (ECG currently has no capacity for carrying out generation). In addition, while ECG has facilities for handling metaphor, these don't enable reasoning-heavy, flexible metaphor processing of the kind which ATT-Meta is capable of, and this is needed to handle the open-endedness of metaphor. However, while the approach taken by ATT-Meta presumes that sufficiently frozen conventional metaphorical phraseology can be handled by fairly straightforward lexical access, bypassing ATT-Meta's use of reasoning and mappings, no facilities for such lexical access are available in the implemented ATT-Meta *system*; as a comprehensive cognitive linguistics account of natural language, ECG, on the other hand, is well-equipped to

[4] Meaning in ECG is grounded in an external ontology, see [27] for details.

[5] It should be noted that our presentation of the ECG formalism here is somewhat simplified, for ease of exposition. For details, see [8], [9], [16], among others.

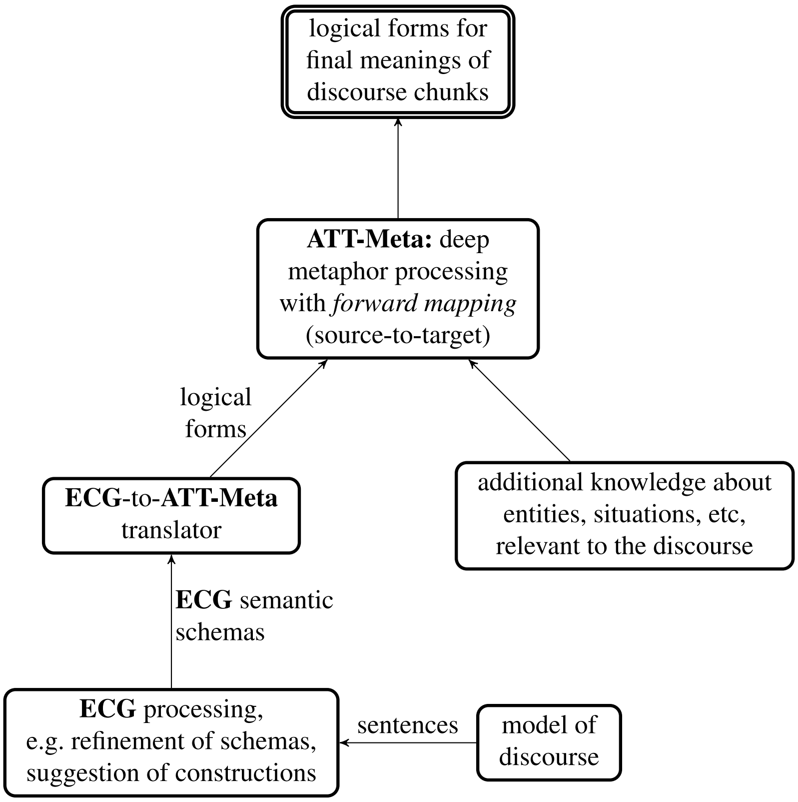


Fig. 1. Schematic of Gen-Meta system: Understanding (stages in system development shown in single-lined boxes, output in double-lined box, selected arrows are labeled with material being piped from one stage to next)

model the richness and variety of metaphorical forms of linguistic expression. Further, while ATT-Meta can indeed model metaphor understanding, the equally important task of metaphor identification is addressed in corpus work we are undertaking ([18], more on this below). Moreover, as also discussed below, we employ computational resources made available via the *ECG Workbench*[6], in order to label our corpora for conventional metaphorical phraseology, as well as to assist in aspects of metaphor identification, and also during understanding.

Diagrammatic representations of both (i) the understanding sub-system, and (ii) the generation subsystem, are presented in Figures (1) and (2), respectively, in line with the discussion in this section.

3.3. Case study: interfacing ATT-Meta and ECG

Consider Figure (3), showing how the sentence *Bill gave Bob a cold* is differently represented by each module (recalling the discussion in section (3.2), in terms of a concrete example): starting with the Goal Logical Form, assuming ATT-Meta has generated it via reverse transfer operations associated with the metaphorical view of DISEASE AS PHYSICAL ENTITY (recall a simpler case in section (2.3.2)), this is then encoded as an appropriate Goal ECG Schema. At this point, the missing ECG generation component presents a gap in our NLG pipeline, to be filled by the DS generation mechanism, by encoding the Goal Schema as a Goal Tree, then proceeding to realise this content as an appropriate surface string. Based on corpus studies (see section (3.4) for more details), specifications for what counts as an appropriate way of realising specific metaphorical content are formulated for the system, and then used to guide realisation toward more natural seeming ways of expressing a chosen metaphor.

The initial stages in our system involve interfacing ECG and ATT-Meta. While our eventual aim is for generation of metaphor, we facilitate development of this interface, by running the system in the reverse di-

[6]http://www1.icsi.berkeley.edu/ lucag/

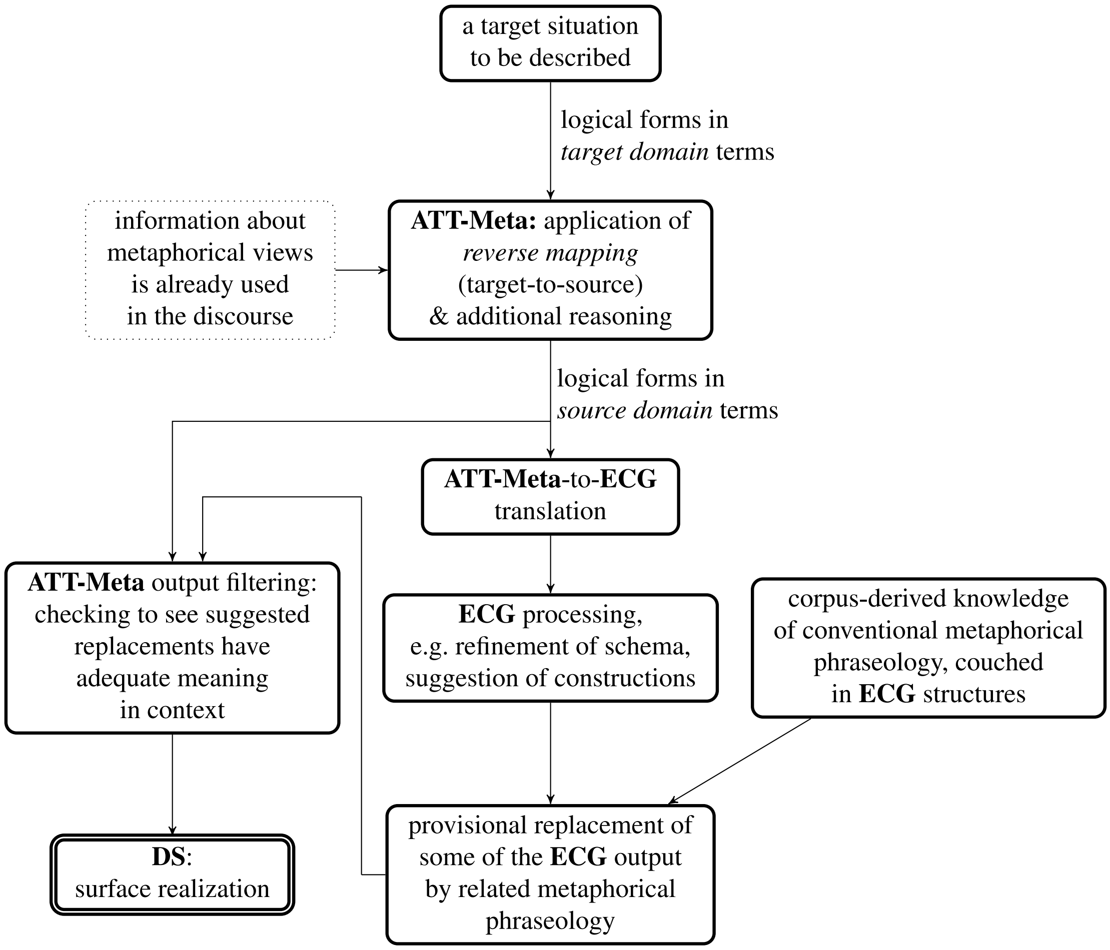


Fig. 2. Schematic of Gen-Meta system: Generation (stages in system development shown in solid single-lined boxes, output in double-lined box, dashed-lined boxes contain additional notes about a particular stage, selected arrows are labeled with material being piped from one stage to next)

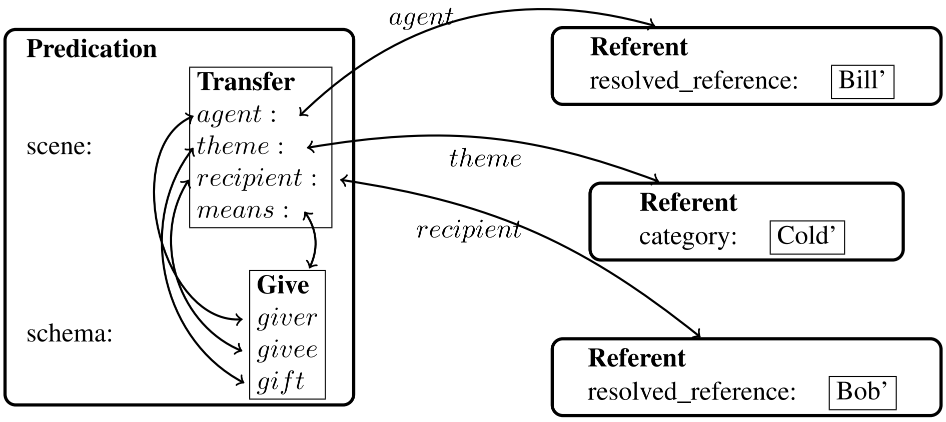
rection as it were, and modeling metaphor understanding by parsing metaphorical utterances using ECG, then piping the output semantic representations to the ATT-Meta module. As both ATT-Meta and ECG are already set up for running in the understanding direction, then this initial step is a shortcut to building the representations required for interfacing these modules.

On the ECG side, carrying out this first step involves initiating an ECG grammar tailored to our corpus. Given the complexity of such a grammar, which involves parsing with respect to both conceptual and linguistic levels (see [16] for details), we will not go into the details of this here. By way of illustration, recall our running example *Bill gave Bob a cold*. A small sample of possible ECG constructions and schemas for parsing just the verb *gave* from this example are provided in Figure (4), and although this selection is somewhat simplified (see [8] for details), important aspects of the ECG approach are represented here. For instance, as [9] notes, there are four ways of specifying relations between ECG schemas and constructions: accessing other structures via roles, inheritance through sub-typing (using the key phrase **subcase of**), evoking another structure (using the key phrase **evokes**), and constraints. All of these ways of specifying such relations are represented here, for example, the VERB construction **evokes** the Predication schema, which is to say “makes it locally available without imposing a part-of or sub-type relation between the evoking structure and the evoked structure” [9].

A further dimension for this fragment of ECG grammar, is that the meaning of the GAVE construction has as its meaning the ObjectTransfer schema (not show here), and one of the constraints on this schema is

GOAL LOGICAL FORM: $\mathbf{ep}(phys\_transferring, bill, bob, cold)$

GOAL SCHEMA:



GOAL TREE:

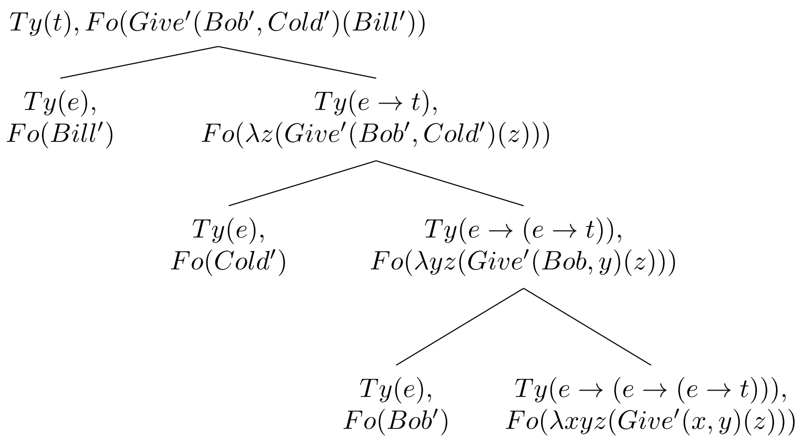


Fig. 3. Representations of the goal content for "Bill gave Bob a cold", for each of the Gen-Meta modules

that the theme (i.e. the thing being transferred) should be a physical entity. Now, of course a cold is a disease but not a physical entity, so that as it stands trying to parse *Bill gave Bob a cold* will fail. However, the utterance is completely understandable, albeit as a metaphorical expression, and an adequate grammar should be able to cover such cases. Fortunately, ECG has the resources available to handle such violations of selection restrictions (a type of collocation restriction) that metaphor can introduce, and so enable the parsing of metaphorical expressions. Our strategy is to make the basic ECG grammar reflect the metaphorical mapping DISEASE AS PHYSICAL ENTITY by extending this grammar with structures for analysing metaphorical expressions.

Consider a simplified version of such an extension specifically for our running example, which is set out in Figure (5). Now, an additional aspect of this representation which has not as yet been discussed is ECG's use of ontological categories [27], such as @*physicalEntity* and @*disease*; in order to enable the metaphorical mapping we are considering here, an additional ontological category is required: @*diseaseAsPhysicalEntity*, which is in fact a subtype of @*metaphors*. To see the impact of this, note that the meaning of the **MetaphoricalNPDPE** construction is the DPEMetaphor schema, but its parent construction is METAPHORICALNP, and following a complicated series of inheritance relations, including *target* and *source* roles being bound under the constraints @*physicalEntity* and @*source*, respectively, it can be shown this proceeds via the mapping licensed by the @*diseaseAsPhysicalEntity* category. Importantly, taking this approach allows us to successfully parse metaphorical utterances, by making a rather minimal extension to a simple ECG grammar.

With this grammar in hand, we are in a position to annotate our corpus with ECG-based schemas and constructions, and thereby make this information available to the ATT-Meta module. To proceed with this, we require a way of capturing such information, and for this we have utilised the ECG Workbench tools.[7] Employing these tools enables a sentence to be parsed in ECG terms, with resulting output being a so-called *semspec*, an elaborate feature structure combining constructions and schemas for each parsed utterance. This semspec can then be processed via the Workbench's *Specializer* tool, and any and all information deemed relevant to our purposes can thereby be extracted.

Most importantly, if ECG has identified a sentence as containing a metaphorical construction such as MetaphoricalNPDPE, then this identification tells ATT-Meta that it can put the literal meaning of the sentence (found by ECG) in a pretence cocoon. Additionally, assuming the ECG ontological category @*diseaseAsPhysicalEntity* from the DPEMetaphor schema is linked to ATT-Meta's mapping rules such as **(R3)**, ATT-Meta has a head start on knowing which mapping rules are relevant (something which otherwise ATT-Meta is required to establish for itself by reasoning steps). Note that if ECG itself comes up with an adequate metaphorical meaning for the sentence, then ATT-Meta's processing can be bypassed. However, if there are non-conventional elements of the sentence (as determined by corpus-derived knowledge) that are not handled by ECG then these can be used to prevent that bypassing.

Also, suppose previous sentences have already stimulated the use of the above ontological category or ATT-Meta's disease-related mapping rules. Then for the current sentence these rules, the literal meaning of the sentence, and the pretence cocoon can be brought into play even if the this sentence is wholly conventional. This caters for cases where a metaphor is reenlivened by discourse context. It allows the literal meaning of the current sentence to contribute to the building of a fuller, more coherent source scenario in the pretence cocoon than would be possible if the current sentence were simply given its own metaphorical meaning, with the literal meaning being thrown away.

An advantage Gen-Meta has over other NLG approaches, is that reasoning done by the AI module increases overall system flexibility. Thus, if it turned out that, following example (6), Bill is no longer infected by a cold, ATT-Meta's reasoning about such change in circumstances, plus ATT-Meta's reverse transfer ability, can lead to the conclusion in the pretence that John physically lost the cold. This proposition can then be piped to the ECG module. This greater control over content specification also has the potential to directly address so-called *strategic generation*, a relatively under-researched area of NLG [12]. Further, the data-driven aspect of Gen-Meta, means that candidate expressions are favored which more closely match formulaic expression of metaphor: so that the relatively formulaic sentence *Bill gave a cold to Bob* would be favored over something as novel as *Bill foisted a cold on Bob*.

3.4. Empirical investigation of metaphor in illness discourse

Our combined AI/corpus-based approach enables fine-tuning of (tactical) generation by clothing AI-generated content in patterns of typical metaphorical expression, as determined via corpus-based discovery of conventional forms of expression. To this end, we have mined such web-based sources as online discussion forums, within one of our chosen domains of illness discourse.[8] Work on illness metaphors is long established, with [30] listing the following examples (among others): ILLNESS AS PUZZLE (e.g. *diabetes is*

[7] http://www1.icsi.berkeley.edu/ lucag/

[8] We are currently analysing a corpus we have collected of our other chosen domain, political conflict discourse, with a view to replicating the approach we have used for illness discourse.

schema Predication
  roles :
    scene
    schema
    event-structure
    setting

construction VERB
  form : $Word$
  meaning :
    evokes Predication **as** pred

construction PRED-EXPR
  form : $Schematic - Form$
  meaning : $Predication$

construction GAVE
  subcase of VERB
  form
    phon : /gæɪv/
  meaning : $ObjectTransfer$
    $self_m \longleftrightarrow$ pred.schema
    pred.setting.time $\leftarrow$ past

Fig. 4. Selected ECG constructions and schemas required to parse the verb token "gave"

a problem to solve), BODY AS MACHINE (e.g. *your body repairs itself*), DOCTOR AS CONTROLLER (e.g. *my GP is trying to control my disease*), ILLNESS AS ATTACK (e.g. *an asthma attack*).

We have collected a corpus of online discussion forums for illnesses of various kinds, including diabetes, stress, infections and cancer, and we annotated illness metaphors in this corpus (see [18], for more details). These annotations yielded metaphor types such as those reported in [30], as well as novel modifications of these types, such as PATIENT AS CONTROLLER (e.g. *What most people do to control type 2 diabetes actually makes their blood sugar get worse!*),[9] as well as completely fresh metaphors, such as what might be dubbed ILLNESS AS RIDE (e.g. *the diabetes rollercoaster*).[10] Table 2 reports initial results for frequency of metaphor types for different illnesses. A Pearson chi-squared test on this data yields $\chi^2 = 125.8$ ($p < .005, df = 9$), suggesting metaphor is not independent of domain of illness.[11] These results are interesting in their own right (see [18]), yet can also be exploited to fine-tune Gen-Meta output. Roughly, relative size of the value in a cell in this table (indicated by standardized residuals in brackets) suggests relative contribution to the overall chi-squared value. For example, comparing standardized residuals for table cells, we could say that while we can be confident that a natural-seeming metaphor about stress is ILLNESS AS ATTACK (e.g. *stress attack* is quite common), this is not the case for diabetes (e.g. *diabetic attack* is far less common). Future work will refine how to best integrate such empirical findings for improving performance of our system.

3.5. Plans for evaluating output of the system

Although our system is still at the development stage, we are already undertaking steps toward building a framework for evaluating the metaphors outputted by the generation system. In particular, we have both subjective and objective measures planned. For this, we have drawn on a range of work evaluating metaphorical expressions (e.g. [2], [22]).

On the subjective side, we are piloting a survey tool for assessing a range of possible evaluations of the metaphors outputted by the system, including evaluations using descriptors, such as the following: how appropriate (e.g. "How appropriate is this metaphor for your conception of diabetes?"), how vivid (e.g. "How vivid is this metaphor for your conception of diabetes?"), how strong (e.g. "How strong is this metaphor for your conception of diabetes?"), how striking (e.g. "How striking is this metaphor for your conception of diabetes?"), how interesting (e.g. "How interesting is this metaphor for your conception of diabetes?"), how engaging (e.g. "How engaging is this metaphor for your conception of diabetes?"), how clear (e.g. "How clear is this metaphor for your conception of diabetes?"). While we will pilot these items directly, as part of our scale development, and in order to carry out external validation of our scale, we are also planning to re-implement the scale-validation study reported in [22], for each of the final set of descriptors we select for our scale.

On the objective side, coming up with metrics that can measure the quality of automatically generated text, while capturing variability as well as being con-

[9]Evoking the notion of "metaphor tuning" from [10].

[10]For discussion of issues we encountered while labeling metaphors, see [18].

[11]With H_0 "metaphor is independent of domain of illness."

Table 2

Frequency of metaphor types for different illnesses (including standardised residuals in brackets), in online discussion forums (see text for details). Key to types of metaphor: **A** = BODY AS MACHINE, **B** = DOCTOR AS CONTROLLER, **C** = PATIENT AS CONTROLLER, **D** = ILLNESS AS ATTACK.

Metaphor	Diabetes	Infection	Cancer	Stress	ROWS
A	28(-.4)	9(2.3)	11(2.5)	5(-2.3)	53
B	3(.5)	0(-.6)	1(1.)	0(-1.)	4
C	117(3.8)	4(-2.2)	7(-1.9)	18(-3.1)	146
D	8(-5.2)	9(1.4)	8(.4)	47(6.7)	72
COLUMNS	156	22	27	70	**275**

sistent with human judgments, is difficult ([6], [7]). For our purposes, this is made all the more difficult by the nature of our task being to evaluate a rather complex and creative linguistic phenomena in metaphorical expression. We plan a range of possible approaches, including human assessments of system outputs, as well as lab-based evaluations, and impact on natural language task performance (e.g. whether the metaphors we produce are coherent enough for someone to be able to extend or mix them with others, a well-known but nevertheless highly creative human linguistic ability).

4. Conclusion

Our generation approach combines AI techniques for producing metaphorical meanings, with corpus-based approaches for identifying conventionalised forms of metaphorical expressions. This enables three main advantages over existing approaches: (1) compared to other NLG approaches, Gen-Meta combines deep AI reasoning to increase flexibility in generating underlying content, (2) which together with data-driven techniques enables realization to favor formulaic expression of metaphor, and, finally, (3) the greater control over content specification which Gen-Meta affords suggests a new and exciting direction to follow in the under-researched area of *strategic generation* [12].

Acknowledgment

We acknowledge financial support through a Marie Curie International Incoming Fellowship (project 330569) awarded to both authors (A.G. as fellow, J.B. as P.I.).

References

[1] K. Abe, K. Sakamoto, and M. Nakagawa, A computational model of the metaphor generation process, *Cognitive Science Society* (2006), 937-–942.

[2] J.L Alty, R.P. Knott, B. Anderson and M. Smyth, A framework for engineering metaphor at the user interface, *Interacting with Computers*, **13(2)** (2000), 301–322.

[3] S. Bangalore and O. Rambow, Corpus-based lexical choice in natural language generation, *38th Annual Meeting of the Association for Computational Linguistics, Hong Kong, China, October 1-8* (2000).

[4] J.A. Barnden, Metaphor and artificial intelligence: Why they matter to each other, *The Cambridge Handbook of Metaphor and Thought*, R. Gibbs, ed., Cambridge University Press, Cambridge, 2008, pp. 311—338.

[5] J.A. Barnden, Metaphor and context: A perspective from articial intelligence, *Metaphor and Discourse*, A. Musolff and J. Zinken, eds, Palgrave Macmillan, 2009, pp. 79—94.

[6] A. Belz and A. Gatt, Intrinsic vs. extrinsic evaluation measures for referring expression generation, *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, Association for Computational Linguistics, 2008, pp. 197–200.

[7] A. Stent, M. Marge, and M. Singhai, Evaluating evaluation methods for generation in the presence of variation, *Computational Linguistics and Intelligent Text Processing*, Springer, 2005, pp. 341–351.

[8] B. Bergen and N. Chang, Embodied construction grammar in simulation-based language understanding, *Construction Grammars: Cognitive grounding and theoretical extensions*, Jan-Ola Östman and Miriam Fried, eds, John Benjamins, 2005.

[9] J.E. Bryant, *Best-fit constructional analysis*, Ph.D. Dissertation, University of California, Berkeley, 2008.

[10] L. Cameron, Metaphor and talk, *The Cambridge Hand-book of Metaphor and Thought*, R. Gibbs, ed., Cambridge University Press, Cambridge, 2008, pp. 197-–211.

[11] W. Croft and D.A. Cruse, *Cognitive Linguistics*, Cambridge University Press, Cambridge, 2004.

[12] R. Dale and E. Reiter, *Building natural language generation systems*, Cambridge University Press, Cambridge, 2000.

[13] A. Deignan, Corpus linguistics and metaphor, *The Cambridge Hand-book of Metaphor and Thought*, R. Gibbs, ed., Cambridge University Press, Cambridge, 2008, pp. 280-–294.

construction METAPHORICALNP
  subcase of NP
  constructional
    constituents
    core: NP
  meaning : $MetaphorSchema$

construction METAPHORICALNPDPE
  subcase of METAPHORICALNP
  meaning
    $constraints$
      $\textbf{self}_m$.name ← DPEMetaphor
      $\textbf{core}_m$.ont-cat ← @disease
      $\textbf{self}_m$.ont-cat ← @physicalEntity

schema DPEMetaphor
  subcase of $MetaphorSchema$
  roles :
    name : @diseaseAsPhysicalEntity
    target : @physicalEntity
    source : @disease

schema MetaphorSchema
  roles :
    name : @metaphors
    target : @entity
    source : @entity

Fig. 5. Selected ECG constructions and schemas required to parse metaphorical uses of the verb token “gave” (*ont-cat*=“ontological-category”)

[14] D. Dobrovol’skij and E. Piirainen, *Figurative Language: Cross-cultural and Cross-linguistic Perspectives*, Elsevier Ltd, 2005.

[15] D. Fass, Review of A computational model of metaphor interpretation by James H. Martin, *Computational Linguistics*, **17(1)** (1991), 106-–110.

[16] J. Feldman, Embodied language, best-fit analysis, and formal compositionality, *Physics of Life Reviews*, 7(4) (2010), 385-–410.

[17] F. Feldman, E. Dodge, and J. Bryant, A Neural Theory of Language and Embodied Construction Grammar, *The Oxford Handbook of Linguistic Analysis*, B. Heine and H. Narrog, eds, Oxford University Press, 2010, 111–139.

[18] A. Gargett and J. Barnden, Mining online discussion forums for illness metaphors, *LREC*, 2014.

[19] R. Hervás, R.P. Costa, H. Costa, P. Gervás, F.C. Pereira, Enrichment of automatically generated texts using metaphor, *MICAI 2007: Advances in Artificial Intelligence*, A. Gelbukh and Á. Kuri Morales, eds, Springer Berlin Heidelberg, 2007, 944-–954.

[20] A. Idström and E. Piirainen, *Endangered Metaphors*, John Benjamins, 2012.

[21] M.A. Jones, Generating a specific class of metaphors, *Proceedings of the 30th annual meeting on Association for Computational Linguistics, Stroudsburg, PA, USA*, Association for Computational Linguistics, 1992, 321-–323.

[22] M. Sobol-Kwapinska and C. Nosal, How does one conceive time? Measurement by means of Time Metaphors Questionnaire, *Polish Psychological Bulletin*, **40(3)** (2010), 113–155.

[23] G. Lakoff and M. Johnson, *Metaphors We Live By*, University of Chicago, 1980.

[24] I. Langkilde, Forest-based statistical sentence generation, *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference, Stroudsburg, PA, USA*, Association for Computational Linguistics, 2000, 170-–177.

[25] O. Lemon, Learning what to say and how to say it: Joint optimisation of spoken dialogue management and natural language generation, *Computer Speech & Language* **25(2)** (2011), 210-–221.

[26] J.H. Martin, *A computational theory of metaphor*, Technical Report, University of California at Berkeley, 1988.

[27] J. Oliva, J. Feldman, L. Gilardi and E. Dodge, Ontology Driven Contextual Best Fit in Embodied Construction Grammar, *CSLP, Lecture Notes in Computer Science, volume 8114*, Springer, 2012.

[28] M. Poesio and H. Rieser, Completions, Coordination, and Alignment in Dialogue, *Discourse & Dialogue*, 1(1), 2010.

[29] M. Purver, R. Cann, and R. Kempson, Grammars as parsers: Meeting the dialogue challenge, *Research on Language and Computation* **4(2—3)** (2006),289-–326.

[30] J.R. Skelton, A.M. Wearn, and F.R. Hobbs, A concordance-based study of metaphoric expressions used by general practitioners and patients in consultation, *The British Journal of General Practice* **52(475)** (2002), 114–118.

[31] T. Veale and Y. Hao, Comprehending and generating apt metaphors: a web-driven, case-based approach to figurative language, *Proceedings of the 22nd national conference on Artificial intelligence - Volume 2*, AAAI Press, 2007, 1471-–1476.

[32] C. Su and C. Zhou, Constraints for Automated Generating Chinese Metaphors, *Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05) Volume 2*, IEEE Computer Society, 2005, 367–370.