Using an Independent, Belief-Based Agent Modelling Component for Language Generation: A Feasibility Study

Abstract

This paper generalises previous work in order to establish desiderata for an independent user modelling component to be used for language generation. Within this context, we discuss the suitability of ViewGen, a powerful belief modelling system and its advatanges for generation. We then conclude by outlining some future work and problems which might arise.

1 Introduction

The field of Natural Language Generation (NLG) studies the automatic production of coherent texts from some internal representation. Often these texts are made sensitive to the context and the target audience, e.g., explanations for novices should contain different information from explanations produced for experts. In this paper we will focus mostly on what features of the user have been needed by previous NLG systems and what other related problems have been tackled (Section 2). Then, we will use this as a basis for establishing the desiderata for a specialised belief modelling component if it is to be used for language generation. Based on that, an overview of ViewGen, a particular belief modelling system, will be given (Section 3). Section 4 then discusses how such a model can influence the generation process and what are the advantages offered by ViewGen in that respect.

2 Previous Work

This section will provide a summary of the type of information which has been encoded in user models in various NLG systems; how it is acquired and how it is updated and maintained. To this end, we generalised the approaches taken in the following systems: PAULINE (Hovy, 1988), TAILOR (Paris, 1993), EPICURE (Dale, 1992), FN (Reiter, 1990), KAMP (Appelt, 1985), ADVISOR (McKeown et al., 1985), ROMPER (McCoy, 1988), PEA (Moore, 1995), EDGE (Cawsey, 1992), WISHFUL (Zuckerman and McConachy, 1993).

2.1 What is modelled?

The different NLG systems need a versatile user-related information in order to provide tailored utterances. The main differences seem to come from (*i*) system task (e.g., advice-giving) and domain; (*ii*) means of communication with the user (e.g., interaction and modality); (*iii*) generation issues addressed; (*iv*) methods and approaches.

2.1.1 Modelling knowledge and belief

Many systems have attempted to track user's *knowledge level* (e.g., PAULINE, TAILOR, EDGE (even updated dynamically)) and *known concepts and facts* – ROMPER, PEA, TAILOR, WISHFUL (concepts and *relations*, including origin of knowledge), KAMP. This also includes *domain taxonomy* (EPICURE, ROMPER); *instances of topics and subtopics* (EDGE (corresponding to instantiated content planning rules)); *typical attributes of classes* (FN); *mutual beliefs* (KAMP); *beliefs about other agent's beliefs and goals* (KAMP); and *inferences* which the user is likely to make (WISHFUL (including confidence levels)).

In addition, PAULINE explores the influence of pragmatic information on generation decisions by discrete modelling of user's social and personal aspects such as *interest*, *opinion* and *emotional state*.

Another important aspect of user's knowledge concerns *language use* (PAULINE). Also, as modelled in the FN system, some domain concepts may lack corresponding lexical realisations – a distinction which might be needed within a multilingual setting too. In addition, FN models *basic level classes*, i.e., cognitively preferred classes which are often used in referring expressions (e.g., use of *dog* instead of a *animal* or *bulldog*).

2.1.2 Goals and plans

Further constraints on tailoring and interpretation of user's responses might come from user's *goals and plans*. There are various types of goals some of which are *social* (e.g., interpersonal PAULINE) while others are *domain- and task-related* – ADVISOR (goals inferred from discourse and plans). Some of these goals may not even have corresponding immediate plans, e.g., *high-level goals* in PEA. Related to that is modelling of user's knowledge of *methods* for achieving goals and performing acts (PEA, EPICURE).

Still another distinction that needs to be made is between *short-term* and *long-term* goals (PAULINE). The former can be achieved and discarded while the latter are never fully achieved and influence various generation decisions without having respective plans.

2.1.3 Separation of models

An important problem which often arises especially in tutoring and advisory systems is allowing for *differences* between the knowledge of the different agents (usually system and user). These differences can be in *user characteristics* (PAULINE); *domain knowledge* (ROMPER); and even *domain ontology or taxonomy* (ROMPER). This separation also requires special reasoning mechanisms which can handle conflict resolution.

2.1.4 Unreliability of the information

Some systems (e.g. EDGE, WISHFUL) also attach *certainty values* to each UM entry in order to compensate for the unreliability of the acquired and inferred information. This information influences in turn the behaviour of the system, e.g., generation of meta-comments such as '*As you probably know,...*' (WISHFUL).

Alternatively, unreliability can be compensated by allowing follow-up questions and clarifications (e.g., PEA).

2.1.5 Deciding what is relevant

More complex domains and richer knowledge bases require context-sensitive mechanisms in order to select relevant knowledge or make an appropriate inference. Therefore, some systems rely on *domain perspectives* or *viewpoints* which affect salience and object similarity. ROMPER maintains *static* perspectives which are orthogonal to the ontology and are used to determine salient object attributes and similarities. A step further towards the dynamic maintenance of perspective is made in ADVISOR, where the active perspective is inferred from user's utterances. These works are further extended in PEA where perspective is also determined from the system's previous utterances by utilising the available problem-solving knowledge.

2.2 Initial Acquisition

The initialisation of the user models very often relies on stereotypical information which is ascribed to the particular user as soon as appropriate stereotype is detected (some systems allow more than one stereotype to hold) (e.g., PEA, EDGE). Other sources of information come from *observing the user* prior to and during the interaction. For instance, PEA analyses user's code, initial request and explicitly stated high-level goals.

2.3 Update and Maintenance

Many NLG systems rely on other existing components to update and maintain the user model (e.g., an understanding component). Nevertheless, some do apply certain techniques to infer user-related information from the ongoing interaction. Some of the solutions are tied to the particular domain or task at hand, while others rely mostly on natural language communication.

The most simple techniques *update* user's domain knowledge by adding all newly introduced concepts (e.g., TAILOR) or at least accumulate past exchanges in a *dialogue history* (e.g., PEA). Further updates may come from the effects of realised speech acts and user's answers to explicit questions (e.g., EDGE).

A less reliable information might be acquired by inference from *stereotypes* and previous user answers (e.g., EDGE). The system can also infer a user's knowledge of a concept in the type hierarchy based on the knowledge of its super- and sub-ordinate ones (e.g., EDGE, WISHFUL). *User's goals* can be also updated based on the analysis of a single utterance or a sequence of utterances (ADVISOR).

2.4 Summary

To summarise, many different types of user information have been represented in various NLG systems. However, none of these systems has attempted to use a comprehensive user modelling approach that can handle *all* these phenomena in a computationally feasible manner. Moreover, each system has developed its own user model with its own representation and expressiveness. In our view, what is needed is an independent, powerful agent modelling component which can be used to support both understanding and generation. The advantage of using such a component is that potentially it could be reused between application domains and also between generators. Therefore, we will now examine an existing agent modelling approach which we intend to integrate into a language generation system.

3 Modelling Agents Attitudes – ViewGen

ViewGen (Ballim and Wilks, 1991a; Lee and Wilks, 1996; Lee and Wilks, 1997) is an agent modelling system which represents beliefs, intentions and goals of dialogue participants. It represents such attitudes in *nested environments* (see Figure 1). There are two types of environments – *viewpoints*, which represent someone's beliefs (e.g., John's beliefs) and *topic* environments, which contain propositions about this topic (e.g., facts about New Mexico). Rather than

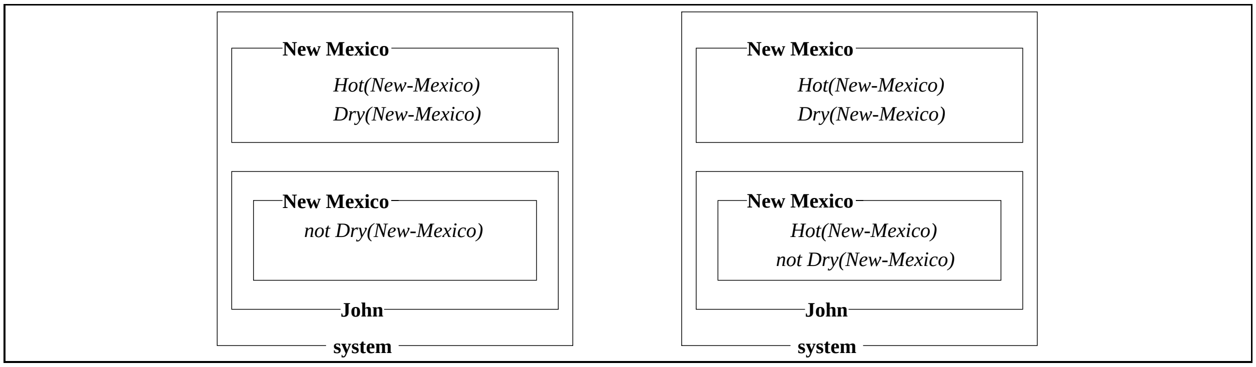

Figure 1: Examples of ViewGen belief environments: before ascription (left) and after ascription (right)

model these statically, ViewGen constructs environments dynamically by a process of *ascription*.

Ascription assumes that attitudes held by one agent can be ascribed to others. There are two main methods of ascription – *default* ascription and *stereotypical* ascription (Ballim and Wilks, 1991b). Default ascription applies to common attitudes which ViewGen assumes that any agent will hold and also ascribe to any other agent *unless there is contrary evidence* (see Figure 1). Stereotypical ascription applies to 'uncommon' attitudes which ViewGen assumes hold only for agents of a particular stereotype.

Both methods are examples of default reasoning and some mechanism is required for dealing with conflicting ascriptions. To handle such problems, ViewGen relies on a truth maintenance system (TMS) (Doyle, 1979). Since ascriptions are heuristic, ViewGen has to be capable of updating its model of the attitudes of the user in the light of newly acquired information. Consequently, the TMS is also used for belief revision.

In brief, ViewGen has all the features which might be required by a language generation system (see previous section) since it:

- can represent beliefs, goals and plans of an agent;
- attitudes are grouped by topic into separate environments which can help reducing the search for relevant information;
- supports nested models which allow representation of conflicting beliefs between agents;
- utilises stereotypes and default ascription to infer additional information;
- resolves conflicts with the help of a truth-maintenance system.

Therefore, we intend to integrate ViewGen into a prototype language generation system in order to assess its suitability as an agent modelling component and identify potential problems of the approach.

4 Generating User-Sensitive Explanations

As it has been already proven in previous research on language generation (e.g., (Paris, 1993)), user-related information could be used to constrain generator's decisions and to improve the fluency and tailoring of the generated text. Such an information is useful at any stage of the generation process:

- *selecting relevant knowledge* – enables explanations[1] with variable detail and information content which also take into account previous explanations and possible user plans and goals;
- *text organisation* – the text organisation strategies or plan operators could have preconditions dependent on user information.
- *surface realisation* – use of words which are known to the user.

In order to restrict the issues down to a manageable task, we intend to study mostly how the agent models in ViewGen could support content selection and text organisation. The resulting language explanations will be generated from that content by an existing realisation component (Bontcheva, 1997).

An important advantage of ViewGen is its account for common knowledge (i.e., default ascription) and stereotypes, as this will allow the modelling of different classes of users and tasks and the use of the most relevant one(s). In addition, the topic environments will restrict the search for relevant information, i.e., they can act as natural *relevant knowledge pools* (McKeown, 1985) from where the text planning process will decide which information to include in the explanation.

[1] Here we will discuss explanations since this is broadly the generation task we are mostly interested in. However, these points are also valid for other generation tasks such as tutorial dialogues.

An essential aspect of ViewGen's paradigm is having a main system's environment which contains environments for system's and agents' attitudes. The consequence of that for the generator's resources is that ViewGen will contain both the domain knowledge base and the user model, while they are usually differentiated in other generation approaches. However we take this to be an advantage since it facilitates the representation and maintenance of conflicting user and system knowledge. Another mostly implementational benefit is the unified interface to the agent modelling component.

The ViewGen agent model will be *updated* after each generated explanation. Since we intend to implement the explanations as hypertext, the user feedback and requests will be always expressed as following this link or another. Since links are not always sufficient to provide the user with effective means of obtaining information, we also plan a context-sensitive questions menu.

Another source of incoming information in hypertext browsing are the bookmarks which, effectively, indicate a user's interests. Therefore, we intend to provide a bookmarking facility which could be used by the system to initialise the user model in subsequent interactions. This and other problems and open issues are discussed in more detail in (Anonymous, 1997).

5 Conclusion

In this paper we have identified a number of user modelling features which are needed by NLG systems in order to enable the production of tailored texts. We have also argued that a language generation system can benefit from the adoption of a separate, specialised agent modelling component that is powerful enough to handle all these phenomena. On behalf of the generation system, this will facilitate reuse across domains and applications, whereas on behalf of the modelling system, this will be a testbed for the feasibility of the approach. ViewGen is such a model which has also been used successfully to deal with a large set of phenomena, e.g., metaphor (Ballim and Wilks, 1991a), conversational implicature (Lee and Wilks, 1997) and reference resolution (Wilks and By, In Press). Therefore, we are now building a system to explore ViewGen's suitability for language generation.

The main problem with nested belief models in generation has proved to be the high cost of use and maintenance (Cahour and Paris, 1991). However, ViewGen's topics and default ascription limit the amount of necessary inference and keep the nestings down to the minimal possible level, which results in a computationally efficient system. However, whether this will be sufficient for real-time generation remains yet to be proven in our future work.

References

Anonymous. 1997. Combining language generation and belief modelling into a flexible hypertext system. In *Proceedings of the Flexible Hypertext Workshop*, Southampton, UK, April. A workshop held in conjunction with the Eight ACM International Hypertext Conference (Hypertext'97).

D. E. Appelt. 1985. *Planning English Sentences.* Cambridge University Press, Cambridge, England.

A. Ballim and Y. Wilks. 1991a. *Artificial Believers.* Lawrence Erlbaum Associates, Hillsdale, New Jersey.

A. Ballim and Y. Wilks. 1991b. Beliefs, stereotypes and dynamic agent modelling. *User Modelling and User-Adapted Interaction*, 1:33 – 65.

K. Bontcheva. 1997. Generation of Multilingual Explanations from Conceptual Graphs. In R. Mitkov and N. Nicolov, editors, *Recent Advances in Natural Language Processing: Selected Papers from RANLP'95*, volume 136 of *Current Issues in Linguistic Theory (CILT).* John Benjamins, Amsterdam/Philadelphia.

B. Cahour and C. Paris. 1991. Role and use of user models. Technical report, ISI/RR-91-323, Information Sciences Institute.

A. Cawsey. 1992. *Explanation and interaction: the computer generation of explanatory dialogues.* Cambridge, Mass.; London:MIT Press.

R. Dale. 1992. *Generating referring expressions constructing descriptions in a domain of objects and processes.* MIT, Cambridge, MA.

J. Doyle. 1979. A truth maintenance system. *Artificial Intelligence*, 12:231 – 272.

E. H. Hovy. 1988. *Generating natural language under pragmatic constraints.* Lawrence Erlbaum, Hillsdale, New Jersey.

M. Lee and Y. Wilks. 1996. An asciption-based approach to Speech Acts. In *Proceedings of the 16th Conference on Computational Linguistics (COLING-96)*, Copenhagen, Denmark.

M. Lee and Y. Wilks. 1997. Eliminating deceptions and mistaken belief to infer conversational implicature. In *IJCAI-97 workshop on Conflict, Cooperation and Collaboration in Dialogue Systems.*

K. F. McCoy. 1988. Reasoning on a dynamically highlighted user model to respond to misconceptions. *Computational Linguistics*, 14 (3), September.

K. R. McKeown, M. Wish, and K. Matthews. 1985. Tailoring explanations for the user. In *International Joint Conference on Artificial Intelligence*, Los Angeles, CA. International Joint Conference on Artificial Intelligence.

K. R. McKeown. 1985. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text.* Cambridge University Press, Cambridge, England.

J. Moore. 1995. *Participating in Explanatory Dialogues.* MIT Press, Cambridge, MA.

C. L. Paris. 1993. *User modelling in text generation.* Francis Pinter Publishers, London.

E. Reiter. 1990. Generating descriptions that exploit a user's domain knowledge. In R. Dale, C. Mellish, and M. Zock, editors, *Current Research in Natural Language Generation.* Academic Press, London.

Y. Wilks and T. By. (In Press). Protocols for reference sharing in a belief ascription model of communication. To apper in *Cognitive Science.*

I. Zuckerman and R. McConachy. 1993. Generating concise discourse that addresses a user's inferences. In *Proceedings of IJCAI'93.*