

Investigating NLG Architectures: taking style into consideration

Daniel S. Paiva

Information Technology Research Institute
University of Brighton
Lewes Road
Brighton BN2 4GJ, UK
Daniel.Paiva@itri.brighton.ac.uk

Abstract

In this paper we propose a methodology for investigating the relationship between architectures of natural language generation (NLG) systems and stylistic properties of texts. Biber's (1988) methodology is used to obtain both the characterisation of style of our corpus and the division of the corpus into sets of linguistically similar texts. These sets will be used for studying the architectural aspects.

1 Introduction

We started our research with a survey of 19 applied natural language generation (NLG) systems (Paiva, 1998) and noticed that:

- almost all the systems followed a pipeline model;
- there was a general agreement on the core NLG tasks that a system should perform (e.g., aggregation, lexicalisation, etc.);
- the surveyed systems mainly differed on the order the NLG tasks are executed (see Cahill et al., 1999);

We also noticed that the texts produced by the various systems were apparently quite different stylistically (although we did not have a formal method to measure how much different they were) and, in order to explain how this variety of texts was obtained with the same type of architecture (i.e., pipeline), we have put forward the hypothesis that the order in which the NLG tasks are executed influences the kind of text that can be obtained, i.e. a certain order would facilitate the generation of a certain type of text whilst another would not.

This hypothesis goes in line with other researchers' results which purport to show that architectural aspects of a NLG system depend on the characteristics of the text to be generated and vice versa. For instance:

- Robin (1994) argues for a revision-incremental architecture for the generation of structurally complex text with floating content — i.e., content that can appear anywhere in the text and is opportunistically realised only if stylistic factors of the surrounding text allow;
- Inui and colleagues (1992) conclude that in order to avoid the generation of text with ambiguous and complex sentences a revision architecture is necessary, with the revision module placed at the end of the generation process (i.e., after the linguistic realisation)¹;
- Reiter (ms.) reports that pipeline architectures cannot deal with constraints on the length of a text.

It is difficult however to reconcile their results in a unified perspective since each of these reported works started with a different perspective and, generally, had different aims in mind.

We believe that it is possible to relate those characteristics of text presented above (such as complexity, ambiguity and sentence length)² to *style*,³ and that we can gain insight into NLG architectures having a systematic way to classify texts by their stylistic properties so that we can analyse the architectural aspects in relation to this stylistic classification.

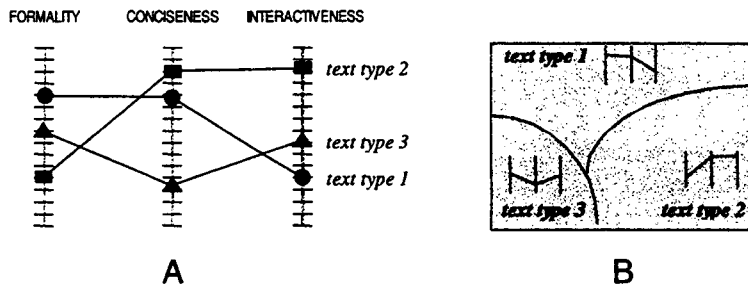
We then start with the point of view that it is reasonable to assume that certain styles of text demand a more specialised type of architecture than others (for example, a revision versus a pipeline architecture⁴) and our idea is to develop a methodology for studying which are the appro-

¹ Some aspects related to the complexity of a sentence (e.g., sentence length) can only be measured precisely when the text has already been generated.

² For instance, complexity and (lack of) ambiguity are factors that can be related to the 'clarity' of a text.

³ Depending on the definition we assume for it.

⁴ For a classification of NLG architectures, see (De Smedt et al., 1996)



In this figure, one group expresses texts that are formal, concise, and not interactive (text type 1 — a possible example is news columns in scientific magazines). Another group (text type 2) expresses texts that are informal, highly concise, and highly interactive (e.g., short articles in TV magazines answering readers' questions). A third one (text type 3) can be considered neither formal nor informal, is not concise and has a medium value for interactivity.

Figure 1: A fictitious example of the characterisation of text types in terms of three stylistic parameters (A), and the partition of the corpus into three text types (B).

priate architectures for the generation of texts in a specific style or more than one style.

Hovy (1988) used a similar approach but characterised style in such an informal way that its relation to architectural aspects was compromised; in particular, he could not ensure that he was not missing important relations between style and generator decisions⁵.

In this paper we will present a characterisation of style and an approach for dealing with it which, we hope, will provide a means to clarify the interaction between the architectures of NLG systems and the type of texts they can, or need to, generate. The paper continues in the following way: in Section 2 we present the definition of style we are working with and in Section 3 we show how this characterisation will help us to deal with aspects of architecture. In Section 4 we discuss the expected results and, finally, we conclude by presenting where we are in this process.

2 Investigating Style

We use the term *style* to signify the variability in the use of features of a language that can be correlated with certain types of situation — where situation can be regarded “as the context within which interaction of ‘the speech event’ occurs” (Brown and Fraser, 1979; p. 34), involving the participants, the setting and the purposes of the communication.

In order to put this definition to work on our problem, first we need to know how to obtain the set of stylistic parameters, i.e., the parameters that, when varied, will be responsible for producing texts in different styles. Secondly, we need to find a way to group stylistically-similar texts into sets so that we can study the architec-

tural aspects of each set isolatedly and then try to have a more general view of how style and NLG architectures interact by making a cross-comparison among those isolated analyses.

For the first part, there are two approaches in the literature linking stylistic parameters to characteristics of texts. We have already cited Hovy (1988) and its main problem: the lack of formality. DiMarco's (1990) approach was to construct a ‘stylistic grammar’ using the notion of norm and deviation from norm (see, e.g., Enkvist (1973)). While this approach is enough to obtain the stylistic parameters⁶, we think that this characterisation brings a problem for grouping texts into sets — it can create only two sets for studying: the texts that agree with the norm and those that do not. This can be a problem because, although the texts following the norm will comprise a set of similar texts, those in the ‘deviant’ set can be so dissimilar that any type of analysis based on them (and, consequently, its interpretation in architectural terms) is probably doomed to failure.

In our approach we will avoid these problems by following a methodology that can provide two things: (1) a characterisation of the stylistic parameters of a corpus⁷, and (2) a partition of a corpus into sets of linguistically similar texts. We are working with Biber's (1988) methodology. From a corpus tagged with a comprehensive set of linguistic features of English, we obtain the frequency count associated with each one.⁸ Using a statistical factor analysis, we group the linguistic features that co-occur, considering each group

⁵ “The specific pragmatic [and stylistic] features used by PAULINE are but a first step. (...) The strategies PAULINE uses to link its pragmatic [and stylistic] features to the actual generator decisions, being dependent on the definitions of the features are equally primitive” (Hovy, 1990; p. 193).

⁶ DiMarco refers to them as ‘stylistic goals’.

⁷ Our corpus comprises texts written for two different audiences (patients and doctors) of more than 250 pharmaceutical products — in total it is more than 500 texts.

⁸ There are two levels of tagging. First, the corpus is tagged using Brill's (1994) tagger. Secondly, programs for counting specific configurations of tags are run. The process is completely automatic.

a stylistic parameter⁹ that can then be analysed in functional terms.¹⁰ Several stylistic parameters can emerge from a corpus, each text of the corpus having a specific value for each of them (see an example with three stylistic parameters in Figure 1-A).

Our interest in Biber's work is also related to his definition of *text types*: "the texts within each type are maximally similar with respect to their linguistic characteristics, while the types are maximally distinct with respect to their linguistic characteristics" (Biber (1995), p. 10). In order to obtain the text types, a cluster analysis is used and results in the partitioning of the corpus (i.e., the texts with similar values for all the stylistic parameters will be grouped in a partition (see Figure 1-B)). Following this procedure will allow us to analyse aspects of architecture for each text type (i.e., each partition) in isolation and, more importantly, make cross-comparisons among these analyses.

3 Relating Style to Architecture

We are using NLG tasks as the basis for our approach to relate style to NLG architectures. We are working with a set of core NLG task that we have found to be stable: all of them occurred in almost all the systems we surveyed (Paiva, 1998). The set comprises the following tasks: content determination, rhetorical structuring, lexicalisation, intra and inter-sentential ordering, referring expression generation, aggregation, segmentation, and linguistic realisation (for an explanation about those task, see Cahill et al., 1999).¹¹

Part of the process for relating style to architecture is depicted in Figure 2. As shown, we start by analysing the NLG tasks that are responsible for the presence of a specific linguistic feature (arrow B). The association of stylistic parameters with linguistic features obtained in the corpus analysis (arrow A) will be used to observe which NLG tasks are responsible for

specific values of a stylistic parameter (arrow C).¹²

Then we will observe the combinations of the NLG tasks in accordance with each text type (partition) obtained in the corpus analysis. This will give us an idea of which NLG tasks are most responsible for (the linguistic features associated with) the different text types; also, it will show us how the tasks are working (because of the links to the linguistic features (see Figure 2)). The result of this process will be sets of NLG tasks for each text type.

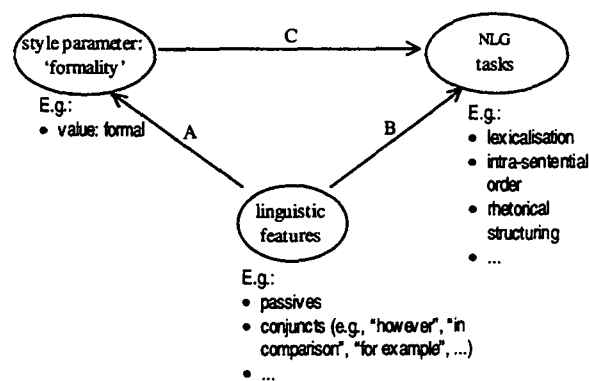


Figure 2: Relating values of style parameters to NLG tasks — a fictitious example supposing 'formality' as a stylistic parameter.

Our work then will be to observe the NLG tasks (inside each text type first, but making cross-comparisons among text types afterwards) attacking the questions related to architecture, i.e., 'which kind of modularisation and interaction between modules is necessary/appropriate', 'which resources are used', 'what kind of data the modules/tasks exchange', etc.

We will investigate architectural decisions at three different levels:

- *at the task level*: how can a certain task be made sensitive to values of stylistic parameters?
- *at the level of tasks interaction*: is there a natural ordering of tasks for a certain type of text?¹³
- *at the global level*: assuming that tasks are normally encapsulated inside modules, what characteristics of texts force the interaction between modules to be more intense?

⁹ Biber refers to this as a 'dimension of register variation'.

¹⁰ The assumption is that strong co-occurrence patterns of linguistic features mark underlying functional dimensions (Biber, 1988; p. 13). Notice that the name of each stylistic parameter, per se, means nothing; it is the linguistic features grouped in each stylistic parameter that are important! Nonetheless, it is easier to refer to a stylistic parameter by its name than to the set of linguistic features it represents. So below we say that a certain text is *formal*, when, in fact, we want to say that *it has certain linguistic features such as passives, formal words, conjuncts, etc.*

¹¹ We are aware that some of those tasks can be subdivided and that some authors assume different names for the same task. If necessary, we will do extensions to this set.

¹² The statistical method by which arrow A in Figure 2 is derived gives a measure of how important the linguistic feature is for a certain stylistic parameter.

¹³ See Danlos (1984) for examples of how the order of execution of tasks can favour a certain textual result over another.

Faced with this classification we will propose solutions that can be used in the specification of an architecture that supports the generation of texts in different styles. We expect these solutions to lead to useful guidelines for helping designers of NLG systems to choose the appropriate architecture for the type of text they want their system to generate.

4 Discussion

One may question why we are repeating Biber's experiment, when he has already obtained a set of stylistic parameters and a set of text types. It is possible that other results emerge from applying his methodology to our corpus, and the only way to know this will be by re-doing the analysis. It is also possible that we obtain a subset of his results, which will at least make our task a more manageable one.

We believe that our result will be of general utility. Although the precise set of stylistic parameters may be dependent on the corpus one is using, we expect that the set of valid task interaction patterns will be restricted, and that the text types emerging from our study will encompass most of the valid patterns. Our programs for counting the linguistic features will be made available for others to use.

5 Conclusion

In this paper we propose a methodology for investigating the relationship between architectures of NLG systems and stylistic properties of the texts they can generate. Although we are still undertaking the first steps of our methodology (the corpus analysis), we believe that this methodology will allow us to test the hypothesis regarding the order of execution of NLG tasks, and at least provide initial indicators on the relationship between classes of architectures and the style of generated texts.

6 Acknowledgements

I would like to thank Donia Scott, Roger Evans, Richard Power, Kees van Deemter, and two anonymous reviewers for their comments on this paper.

7 References

- Biber, D. (1988) *Variation across speech and writing*. Cambridge University Press.
- Biber, D. (1995) *Dimensions of register variation: A cross-linguistic comparison*. Cambridge University Press.
- Brill, E. (1994) Some advances in rule-based part of speech tagging. In *Procs. of AAAI'94*, Seattle.
- Brown and Fraser (1979) Speech as a marker of situation. In (eds.) Shcerer, K.R., and Giles, H., *Social markers in speech*. Cambridge University Press, pp. 33-62.
- Cahill, L., Doran, C., Evans, R., Mellish, C., Paiva, D., Reape, M., Scott, D., and Tipper, N. (1999) In search of a reference architecture for NLG systems. Accepted for EWNLG'99.
- Danlos, L. (1984) Conceptual and Linguistic Decisions in Generation. In *Procs. of the 10th International Conference on Computational Linguistics (COLING'84)*. Stanford University, California, USA, pp. 501-504.
- De Smedt, K., Horacek, H., and Zock, M. (1996) Architectures for Natural Language Generation: Problems and Perspectives. In Adorni, G., and Zock, M. (Eds.), *Trends in Natural Language Generation: an Artificial Intelligence Perspective*, Springer Verlag, New York.
- DiMarco, C. (1990) *Computational stylistics for natural language translation*. PhD Dissertation. Technical report CSRI-239, University of Toronto.
- Enkvist, N. E. (1973) *Linguistic Stylistics*. Mouton.
- Hovy, E. (1988) *Generating Natural Language under Pragmatic Constraints*. Lawrence Erlbaum Associates.
- Hovy, E. (1990) Pragmatics and Natural Language Generation. *Artificial Intelligence* 43, pp. 153-197.
- Inui, K., Tokunaga, T., and Tanaka, H. (1992) Text Revision: A Model and Its Implementation. In Dale, R., Hovy, E., Rösner, D., and Stock, O., (eds.) *Aspects of Automated Natural Language Generation*, Lectures Notes in Artificial Intelligence 587, Springer-Verlag.
- Paiva, D. (1998) A Survey of Applied Natural Language Generation Systems. Technical report ITRI-98-03, Information Technology Research Institute (ITRI), University of Brighton.
- Reiter, E. (ms.) A Problem with Pipelines. Dept. of Computer Science. University of Aberdeen.
- Robin, J. (1994) *Revision-based generation of natural language summaries providing historical background: corpus-based analysis, design, implementation and evaluation*. Ph.D. Thesis. CUCS-034-94, Columbia University.