Generation of Air Pollutant Information

— Project Presentation Note —

Leo Wanner, Bernd Bohnet and Ciprian Gerstenberger
Computer Science Department, University of Stuttgart
Breitwiesenstr. 20-22, 70565 Stuttgart, Germany
{wanner|bohnet|gerstecn}@informatik.uni-stuttgart.de

Abstract In this note, we present a project (*Auto-Text UIS*) on applied text generation. The goal of Auto-Text is the production of air pollutant information from data in a relational data base. Auto-Text consists of two modules: the preprocessor, which evaluates the data compiling from them conceptual structures, and the text generator, which takes conceptual structures as input producing from them the multiparagraph reports. The linguistic framework that underlies the generator is the Meaning-Text Theory.

1 *Introduction*

The field of text generation has passed its early stage when toy applications used to demonstrate the feasibility of the respective theoretical approach. During the last decade, the number of practical applications increased considerably. In this note, we describe the presentation of a project on applied text generation—the *Auto-Text UIS*. Auto-Text is a generator for the production of air pollutant information from data collected on a regular basis at more than sixty measuring stations that are distributed over the province of Baden-Württemberg, Germany. Auto-Text is funded by the Ministry of Environment and Traffic, Baden-Württemberg and developed jointly by the Computer Science Department, University of Stuttgart, the Environment Agency, Baden-Württemberg and the UMEG GmbH (a company specialized in the acquisition of environmental data). In its pilot application proposed for presentation, Auto-Text focuses on ozone emission. The pilot application will be fielded at the end of 2000.

2 *Textual Characteristics*

The general discourse structure of air pollutant information is fixed with respect to the types of information communicated and the order in which these types are presented in the text. Therefore, it can be well captured by schemata as introduced into generation by McKeown. The main schema available for the information provided in connection with one specific measuring station looks as follows:

1. Current concentration at the station in question
2. Compare the concentration with the concentration at a reference time point
   2.1. Concentration at a reference time point
   2.2. Compare current concentration to the concentration at reference time point
3. Evaluation of the concentration
4. Relation of the concentration to thresholds
5. Legal regulations/ warnings/ precautions to be taken
6. Regional concentrations
   6.1. Highest concentration
   6.2 Lowest concentration

Each element of the schema constitutes a proposition. In the course of generation, propositions can be aggregated or expanded into several propositions. Between instantiated elements discourse relations in the sense of the *Rhetorical Structure Theory* (RST) are defined. During the presentation, each element and the RST-relations between them will be explained in detail.

3 *Overview of Auto-Text*

Figure 1 shows the architecture of Auto-Text. It consists of two main modules: the Preprocessor and the Generator. The Preprocessor contains three submodules: the XML-Parser (freely available from SUN Inc.), the Compiler, and the Evaluator. The Generator is

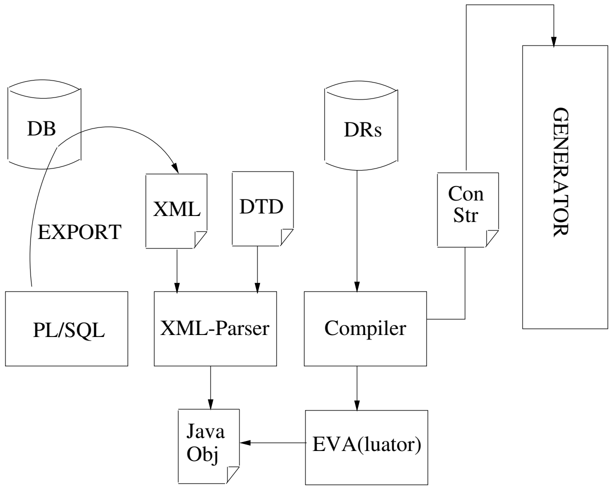

Figure 1: Architecture of the Auto-Text System

based on the *Meaning-Text Theory* (MTT) (Mel'čuk, 1988)—a linguistic theory which becomes increasingly popular in text generation; see, e.g., (Iordanskaja *et al.*, 1992; Lavoie & Rambow, 1997; Coch, 1997). Both the Preprocessor and the Generator are implemented in Java.

As input to the Preprocessor, XML-encoded data from an ORACLE data base are used. The XML-documents are transformed by the XML-Parser into Java internal object structures.

The task of the Compiler is to compile, from the input data, a *Conceptual Representation* (ConR), which serves as input to the Generator. To accomplish this task, the compiler uses two types of domain specific rules (DRs): (i) conceptual rules and (ii) communicative rules.

The conceptual rules derive the *Conceptual Structure* (ConStr) from the input data—a structure which is closely related to Sowa's Conceptual Graphs. The communicative rules superimpose on the ConStr a communicative organization, i.e., a thematic and a rhematic partition, a focus partition, etc. Both types of rules access information from the input data via the Evaluator, which serves thus as an intermediator between the compiler rules and the input data represented as Java object structures. During the presentation, examples of both conceptual and communicative rules will be shown.

Each element within a specific discourse schema is associated with the set of rules which need to be executed in order to compose the fragment of the ConR that realizes this element. A discourse schema is activated by the demand of the reader of information on air pollution in a specific area of the province, on the concentration of a specific substance, on statistics concerning a specific substance, etc.

The Generator starts from a ConR constructed by the Preprocessor. ConR is thus the most abstract stratum of the linguistic description in our generator. In accordance with MTT several further strata are distinguished: semantic (Sem), deep-syntactic (DSynt), surface-syntactic (SSynt), deep-morphological (DMorph) and surface-morphological (SMorph). The generation process consists of a series of structure transitions between adjacent strata until the SMorph stratum is reached. At the SMorph stratum, the structure is a string of linearized word forms. The transitions are carried out by a compiler that maps a structure specified at one of the five first of the above strata on a structure at the adjacent stratum. The presentation will give sample structures of each stratum and examples of rules for each transition.

Further details of the generator will be presented in a demonstration of the development environment which is also used in the Auto-Text project; see (Bohnet *et al.*, 2000).

References

Bohnet, B., A. Langjahr & L. Wanner. 2000. A Development Environment for MTT-Based Sentence Generators. *Proceedings of the XVI SEPLN Conference.* Vigo, Spain.

Coch, J. 1997. Quand l'ordinateur prend la plume : la génération de textes. *Document Numérique.* 1.3.

Iordanskaja, L.N., M. Kim, R. Kittredge, B. Lavoie & A. Polguère. 1992. Generation of Extended Bilingual Statistical Reports. *COLING-92.* 1019–1022. Nantes.

Lavoie, B. & O. Rambow. 1997. A fast and portable realizer for text generation systems. *Proceedings of the Fifth Conference on Applied Natural Language Processing.* Washington, DC.

Mel'čuk, I.A. 1988. *Dependency Syntax: Theory and Practice.* Albany: State University of New York Press.