EasyText: an Operational NLG System

Laurence Danlos
Univ Paris Diderot,
Sorbonne Paris Cité,
ALPAGE, UMR-I 001 INRIA
Danlos@linguist.jussieu.fr

Frédéric Meunier
WatchSystem Assistance
Meunier@watchsystance.com

Vanessa Combet
WatchSystem Assistance
Combet@watchsystance.com

Abstract

This paper introduces EasyText, a fully operational NLG system. This application processes numerical data (in tables) in order to generate specific analytical commentaries of these tables. We start by describing the context of this particular NLG application (communicative goal, user profiles, etc.). We then shortly present the theoretical background which underlies EasyText, before describing its implementation, realization and evaluation.

1 Introduction

EasyText is a NLG system which is operational at Kantar Media, a French subsidiary company of TNS-Sofres[1]. The company compiles numerical data for its customers on their advertising investments and sends to each customer seven tables every month, see Figure 1 for an example of a table. Before the existence of EasyText, these tables were presented with a general commentary written by a media analyst. Kantar decided to accompany these tables with specific charts and commentaries in order to make their reading comfortable and easy. They survey 600 segments and there are 7 tables per segment: Manually writing these analytical commentaries was inconceivable. The idea of having an automatic system producing them naturally arose, but Kantar encountered major difficulties with the text generation task. Therefore, they subcontracted this project to Watch System Assistance. Figure 1 shows an example of an analytical commentary generated by EasyText.

Section 2 describes the architecture of EasyText. Section 3 presents its implementation: EasyText is an instantiation of Kantar Media's needs in a ready-to-use NLG framework, TextElaborator. Section 4 gives some details on the realization and evaluation of EasyText.

2 Architecture of EasyText

EasyText follows a standard architecture as described in (Reiter and Dale, 2000). It includes a document planner for the content determination and document structuring tasks, and a tactical component.

The **content determination** task for a given table amounts to detecting the relevant cells of the table. This task was guided by business rules indicated by Kantar Media's analysts. These rules were hard-coded, i.e. without any reasoning module.

The content of a cell is transformed into the conceptual representation of an eventuality whose predicate is given by the column heading. This predicate subcategorizes two arguments, the first one corresponding to the line heading, the second one to the value of the cell. Therefore, the output of the content determination module can be seen as a conjunction of conceptual representations of eventualities.

The **document structuring** task consists in introducing rhetorical relations between the semantic content of the highlighted cells. For instance, if two opposite evolutions over a given period are observed (one decreasing, the other one increasing), the relation $Contrast$ is introduced. On the contrary, would

[1] Kantar Media is one of the leaders in advertising expenditure monitoring, exploring all existing media (radio, internet, mobile telephony, etc.).

Proceedings of the 13th European Workshop on Natural Language Generation (ENLG), pages 139–144,
Nancy, France, September 2011.

Évolution des investissements par Secteur / Variété

Investissements publicitaires plurimedia - Tri décroissant sur le cumul de l'année en cours - En k€

	Mai 2008	Mai 2009	Evol%	Cumul janvier à mai 2008	Cumul janvier à mai 2009	Evol%
ORGANISMES FINANCIERS	16 587	26 312	59 %	216 948	177 353	-18 %
CREDIT PERSONNEL O.F	5 868	11 227	91 %	50 610	53 772	6 %
~MULTIPROD.ORG.FINANCIERS	3 243	7 463	130 %	53 191	51 718	-3 %
CREDIT RENOUVELABLE O.F	3 930	1 994	-49 %	60 094	34 987	-42 %
INTERNET TELEMATIQUE 583	2 648	4 687	77 %	16 460	27 613	68 %
RACHAT DE CREDITS O.F	777	732	-6 %	15 817	5 637	-64 %
CREDIT AUTO MOTO O.F	79	110	39 %	5 638	993	-82 %
CREDIT TRAVAUX O.F		86		535	797	49 %
PARRAINAGE MECENAT O.F				80	0	-100 %

Dans votre univers, les investissements marquent une très forte progression (+59%) dans le secteur ORGANISMES FINANCIERS en mai 2008 par rapport à mai 2007. Toutefois, pour le cumul à date de l'étude, ils connaissent une baisse de 18%.

Dans ce secteur, les investissements ont doublé (+130%) pour la variété MULTIPROD.ORG.FINANCIERS en mai 2008 par rapport à mai 2007. Par ailleurs, les investissements pour la variété CREDIT PERSONNEL O.F marquent une progression de 6% pour le cumul à date étudié. Au contraire, pour la variété MULTIPROD.ORG.FINANCIERS, ils voient leur volume diminuer (-3%) sur la même période.

(Within your business area, ad spending ramps up (+59%) for sector ORGANISMES FINANCIERS in May 2008 compared with May 2007. However, year to date, it falls 18%. Within this sector, ad spending doubles (+130%) for segment MULTIPROD.ORG.FINANCIERS in May 2008 compared to May 2007. Furthermore, ad spending for segment CREDIT PERSONNEL OF increases of 6% year to date. On the contrary, for segment MULTIPROD.ORG.FINANCIERS, it decreases (-3%) over the same period.)

Figure 1: Example of a table and its automatically generated comment

they have been going in the same direction, the relation $Parallel$ would have been introduced, along with some hints to prepare an aggregation operation in the tactical component.

The discourse theory on which the document structuring module relies is SDRT (Segmented Discourse Representation Theory) (Asher, 1993; Asher and Lascarides, 2001), following (Danlos et al., 2001). The output of the document structuring component is therefore consistent with a SDRS (SDRT structure), considered as a "conceptual" representation in which concepts (discourse relations, eventualities, entities) are embedded in a dependency structure (which is mathematically a Directed Acyclic Graph).

The **tactical component** (macro/micro-planner and surface realizer) is based on G-TAG formalism (Danlos, 2001), the latter being itself founded on lexicalized Tree Adjoining Grammars (TAG), (Joshi, 1985).[2] G-TAG deals with the *How to say it?* issue, understood as covering all (and only) linguistic decisions: segmentation of the text into sentences and linear ordering of these sentences[3], choice of discourse connectives and other lexical items, syntactic constructions within sentences, aggregation operations, referring expressions, semantic and syntactic parallelism, etc.

The surface realizer is designed to use the syntactic and lexical information of a TAG grammar. This TAG grammar is extended to handle multi-sentential texts and not only isolated sentences[4]. Therefore, the macro/mico planner is designed as a TAG extension. More precisely, the architecture of G-TAG is outlined in Figure 2:

- The output of the macro/mico planner is a "g-derivation tree". In TAG, a derivation tree is

[2]Since it was put forward by A. Joshi that TAG is an especially well suited grammatical theory for text generation, adapting TAG for generation has been widely explored, among many others, let us cite (Stone and Doran, 1997) and (Gardent and Kow, 2007). However, it is not in the scope of this paper to compare G-TAG with these other approaches.

[3]These tasks are not considered as part of the document structuring component. This is why the term macro/mico-planner is used in Figure 2.

[4]The idea of extending TAG to handle multi-sentential texts is also used in text interpretation, e.g. D-LTAG (Webber, 2004) and D-STAG (Danlos, 2009).

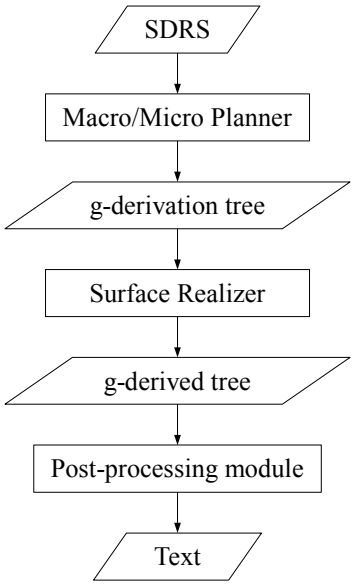

Figure 2: G-TAG tactical component

not only seen as the history of the derivation but also as a linguistic representation, close to semantics, which can serve as a basis for a deeper semantic analysis (Kallmeyer, 2002). A g-derivation tree in G-TAG is closer to semantics than a derivation tree in TAG: it is a semantic dependency tree annotated with syntactic information. Moreover, a g-derivation tree represents a text while a derivation tree represents a unique sentence.

The macro/micro planner relies on lexical databases associated with the various concepts (discourse relations, eventualities, entities) that are relevant for the NLG application. A lexical database for a given element records the lexemes lexicalizing it with their argument structure, and the mappings between the conceptual and semantic arguments. With such a lexicalized planner, the process for computing a g-derivation tree relies upon a single type of operation: lexicalization, i.e. choice of a lexeme and its syntactic realization to convey an instance of a concept. Since all the main decisions are made during this process, G-TAG can be considered as a fully lexicalized formalism for text generation.

- Thanks to a TAG grammar (which specifies the mapping between the semantic and syntactic arguments), a g-derivation tree specifies a unique "g-derived tree", in the same way as a derivation tree specifies a unique derived tree. A g-derived tree is a syntactic tree annotated with morphological information.

- From a g-derived tree, a post-processing module computes a text by performing morphological computations[5] and formatting operations.

Lexical databases for EasyText have been developed by a linguist, V. Combet, who was working in close collaboration with Kantar Media's analysts. Particular attention was paid to linguistic variation in order to avoid producing tiresome texts for Kantar Media's customers. This variation mainly concerns:

- the lexical choices: the databases associated to a given concept are as exhaustive as possible. For example, the concept INCREASE with a MAGNITUDE argument is lexicalized either with the verb *augmenter*, *doubler* or *tripler* or with the light verb construction *être en hausse/augmentation* or *enregistrer une hausse/augmentation*[6]. Moreover, a verb can be modified with an adverb, e.g. *faiblement, fortement, modéremment* for *augmenter* and *presque/pratiquement/plus que* for *doubler* or *tripler*[7], while the noun in a light verb construction can be modified with a preposed adjective, e.g. *faible/forte*, or a postposed one, e.g. *modérée*[8].

- the order of the phrases: some phrases can appear more or less freely in different places in a sentence. This is the case for duration adverbials such as *pendant le mois de mai* (*during May*) and also for different prepositional phrases such as *les investissements [pour la variété X] augmentent [pour la variété X]* (*ad*

[5]Morphological operations include elisions (*la augmentation → l'augmentation*) and contractions (*de le mois → du mois*).

[6]In English, verbs *increase*, *double* or *triple* and light verb constructions *be on the increase* or *record an increase*.

[7]In English, adverbs *slightly, seriously, moderately* for *increase* and *almost, nearly, more than* for *double* or *triple*.

[8]In English, an adjective is always preposed.

spendings [for sector X] increase [for sector X]).

3 Implementation

A prototype of G-TAG was first implemented in Ada (Meunier, 1997). G-TAG has been re-implemented as a ready-to-use framework, TextElaborator. TextElaborator is based on the Microsoft .Net framework. Particular attention was paid on functional and business issues while taking advantage of .Net for technical and non functional issues (persistence, reliability, scalability, etc.). We chose to rely on classical design patterns[9], which garantee an effortless reusability of the different components.

Our main implementation effort for TextElaborator was to build an IDE (Integrated Development Environment) incorporating tools which facilitate the linguistic work, i.e. feeding, editing, debugging and testing the various lexical databases — tasks which are crucial in G-TAG, not only in the development but also the maintenance phases.

A screenshot of this IDE is shown in Figure 3. The left column gives the domain ontology hierarchized as Abstract Objects (discourse relations and eventualities) and Entities. When clicking on a concept of the domain ontology (e.g. `Hausse` in Figure 3), the tab `LexicalPredicates` indicates the G-TAG lexical database associated with this concept. When choosing an element of this database, the corresponding g-derivation tree is displayed, along with some essential corollary information.

EasyText is an instantiation of TextElaborator for Kantar Media's needs, constiting in an ontology and its corresponding lexical databases. TextElaborator is written in C# language and is built and run upon the Microsoft .Net framework. Thanks to .Net, its integration into Kantar Media's information system was easy. Generating a comment as the one shown in Figure 1 requires an average of 400ms.

[9]DAO (Data Access Object, `http://en.wikipedia.org/wiki/Data_access_object`) and DTO (Data Transfer Object, `http://en.wikipedia.org/wiki/Data_transfer_object`).

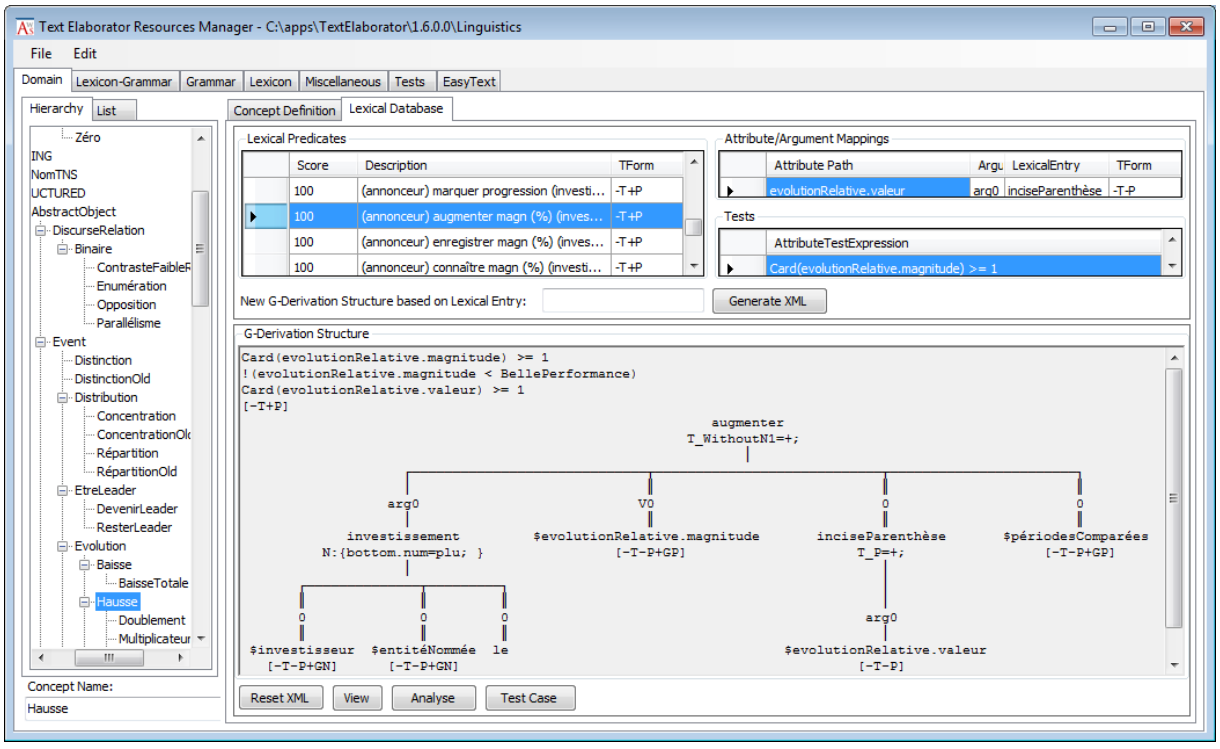

Figure 3: Sceenshot of TextElaborator's integrated development environment

4 Realization and Evaluation

The development of EasyText took 7 mm (men month) altogether:

- 1 mm was dedicated to the linguist's training to TAG and G-TAG;
- 1 mm to interviews with Kantar's media analysts;
- 3 mm to design TextElaborator and its IDE;
- 2 mm to fill the lexical databases.

During these 7 months, we were never in contact with Kantar's customers directly, but worked in close interaction with the two departments involved in the project. On the one hand, we obviously interacted with Kantar's media analysts. They shared with us all their know-how on writing commentaries on Kantar's tables, enabling us to create lexical databases corresponding to their editorial habits.

On the other hand, EasyText was developed in close collaboration with Kantar's Information system department, so as to meet their technical requirements: performance and compatibility with the existing infrastructure.

When we released the first version of EasyText, Kantar decided to send the automatically generated commentaries to a couple of customers, without saying anything about the way they had been written.

These customers made some critics[10] but gave Kantar Media the feedback that they were satisfied with this new offer. Therefore, Kantar Media decided in April 2010 to commercialize this new product and acknowledged that the commentaries were automatically generated. They keep on commercializing it, which seems to mean that their customers are satisfied.

EasyText evaluation was made by Kantar's media analysts during several months. This evaluation was qualitative and concerned the relevance of the commentaries (the choice of the cells to comment) and their accordance to the editorial habits. We remind the reader (Section 1) that EasyText commentaries had never been handwritten. Therefore, we cannot make any comparison between the generated texts and handwritten ones. This situation seems to be common in NLG, since applications are likely to be commercialized when automatic writing doesn't replace hand writing[11]. Indeed, the few commercial NLG systems we are aware of are in the same situation.[12]

[10] The main critic concerned the laying-out of these commentaries.

5 Conclusion

We have presented an operational system and, while many NLG prototypes exist, not many are commercialized, eventhough NLG technology is mature.

EasyText is an instantiation of a ready-to-use framework, TextElaborator, which is based on solid scientific basis concerning not only its architecture — the standard one (Reiter and Dale, 2000) — but also the particular instantiation of this architecture with well-established analysis formalisms (SDRT and TAG) which have been adapted to text generation.

It is foreseen that TextElaborator will be used for other applications and will produce texts in other languages than French, since it was developed as a ready-to-use framework. For a new application, the domain ontology has to be adapted and the G-TAG lexical databases associated with the concepts have to be filled. When moving to another language, only the lexical databases will have to be changed, hopefully.

A demonstration of EasyText will be presented during the conference.

References

Nicholas Asher and Alex Lascarides. 2001. Indirect speech acts. *Synthese*, 128(1–2):183–228.

Nicholas Asher. 1993. *Reference to Abstract Objects in Discourse*. Kluwer, Dordrecht.

Laurence Danlos, Bertrand Gaiffe, and Laurent Roussarie. 2001. Document structuring à la SDRT. In *International workshop on text generation - ACL*, pages 94–102, Toulouse.

Laurence Danlos. 2001. G-TAG: A lexicalized formalism for text generation inspired from TAG. In

[11] Guy Lapalme's personal communication in the late 90's.

[12] See the NLG applications developed in the American business world by Cogintex (`http://www.cogentex.com`) and in the French business world by Yseop (`http://www.yseop.com`).

A. Abeillé and O. Rambow, editors, *TAG Grammar*. CSLI.

Laurence Danlos. 2009. D-STAG: a formalism for discourse analysis based on SDRT and using synchronous TAG. In *Proceedings of the 14th Conference on Formal Grammar (FG'09)*, pages 1–20, Bordeaux, France.

Claire Gardent and Eric Kow. 2007. A symbolic approach to near-deterministic surface realisation using tree adjoining grammar. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 328–335, Prague, Czech Republic, June. Association for Computational Linguistics.

Aravind Joshi. 1985. Tree-adjoining grammars. In D. Dowty, L. Karttunen, and A. Zwicky, editors, *Natural language parsing*, pages 206–250. Cambridge University Press.

Laura Kallmeyer. 2002. Using an enriched tag derivation structure as basis for semantics. In *Proceedings of the TAG+6 Workshop*, pages 101–110, Venice.

Frédéric Meunier. 1997. *Implémentation du formalisme G-TAG*. Thèse de doctorat en informatique, Université Denis Diderot, Paris 7.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge.

Matthew Stone and Christine Doran. 1997. Sentence planning as description using tree adjoining grammar. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 198–205, Madrid, Spain, July. Association for Computational Linguistics.

Bonnie Webber. 2004. D-LTAG: extending lexicalized TAG to discourse. *Cognitive Science*, 28(5):751–779.