Adding communicative structure to the MMT into ACG encoding

Marie Cousin

Adding communicative structure to the MMT into ACG encoding

Marie Cousin
Université de Lorraine, CNRS, Inria, LORIA / F-54000 Nancy, France
marie.cousin@loria.fr

Meaning-text theory (MTT) is a linguistic theory aiming at describing the correspondence between the meaning of an utterance and its surface form. We use abstract categorial grammars (ACGs), a grammatical formalism based on λ-calculus, to implement a version of a model of MTT, from the semantic representation level to the surface syntactic one. This implementation hinges upon abstract categorial grammars composition in order to encode level transitions with transduction operations.

Cousin (2023a,b) offers an encoding of MTT into ACGs, but only takes the predicative structures into account. However, MTT relies heavily on the communicative structures that come along the predicative ones. Theme and rheme play an crucial role in MTT as they determine which deep-syntactic graph will be obtained from the semantic one, e.g., producing a verbal phrase or a noun phrase, such as illustrated in figure 2. This paper extends Cousin (2023a)'s encoding with communicative structure information.

Meaning-Text Theory MTT (Mel'čuk 2012, 2013, 2015) describes the link between the meaning and the textual representation of an utterance. MTT's model (see figure 1) is composed of seven representation layers and six transition modules between each of them. MTT uses, among others, a dependency syntax and the

$$\xrightarrow[\textit{Generation}]{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}$$
$$\text{SemR} \longleftrightarrow \text{DSyntR} \longleftrightarrow \text{SSyntR} \longleftrightarrow \text{DMorphR} \longleftrightarrow \text{SMorphR} \longleftrightarrow \text{DPhonR} \longleftrightarrow \text{SPhonR}$$
$$\xleftarrow[\textit{Analysis}]{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}$$

Figure 1: Hierarchy of MTT's model

key concepts of paraphrases and lexical functions. This paper focus here on the semantic and deep-syntactic representation layers, and the semantic module that bridges them.

Theme and Rheme To all representation levels of MTT correspond several substructures, among which a predicative structure and a communicative structure. The latter adds communicative opposition annotations to the former ones. Theme and rheme are the markers of the thematic opposition in MTT (Polguère 1990). In a sentence, the theme is what we are talking about. It fills the blank in "Speaking of ...", when the rheme is what we are saying about the theme and answers the question "What is said about *theme* ?".

(a) "*Charlie is Taylor's son.*" (b) "*Charlie, the son of Taylor.*"

Figure 2: Illustration of different theme and rheme areas on the same predicative semantic graph (SemS)

The association of a semantic representation (SemR) to a verbal or nominal expression depends on the SemR's theme-rheme partition. For instance, "*Charlie is Taylor's son*" talks about Charlie (the theme), and says that he is Taylor's son (the rheme), while "*Charlie, the son of Taylor*" does not have a communicative opposition.

(a) "*Charlie invites Taylor.*"

(b) "*Taylor is invited by Charlie.*"

Figure 3: Illustration of different theme and rheme areas on the same predicative semantic graph

Another case where thematicity plays a decisive role is when the speaker talks about one specific argument of a semantic predicate (see figure 3). For instance, "*Charlie invites Taylor*" has a different meaning than "*Taylor is invited by Charlie*". The former is talking about Charlie (the theme) who invites Taylor (the rheme) (see figure 3a). The latter is talking about Taylor (the theme), who is invited by Charlie (the rheme) (see figure 3b).

Thus, theme and rheme play a crutial role in the generation process of MTT.

Abstract Categorial Grammars ACGs are a grammatical framework based on λ-calculus. They enable the implementation of other grammatical formalisms and can be used in generation or analysis (Kanazawa 2007), what has proven useful to model MTT (Cousin 2023a).

An ACG (de Groote 2001) is composed of two vocabularies or signatures (an abstract one and an object one) linked together by a lexicon. On the abstract signature, a set of λ-terms can be obtained by induction, the abstract language. It is the set of abstract grammatical structures, such as analysis trees. On the object signature, a set of λ-terms can be obtained by induction, the object language. It is the set of the surface representations generated by the abstract language, such as strings or logical representations in the form of a graph. In figure 4, $\langle\ \Sigma_{\textit{deep-syntacic}}, \Sigma_{\textit{semantic}}, \mathcal{L}_{\textit{sem}}\ \rangle$ is an ACG, with $\Sigma_{\textit{deep-syntacic}}$ its abstract signature, and $\Sigma_{\textit{semantic}}$ its object signature. Three operations are defined over ACGs (see figure 4), application (of a lexicon), parsing (reverse application of a lexicon), and transduction (composition of parsing and application).

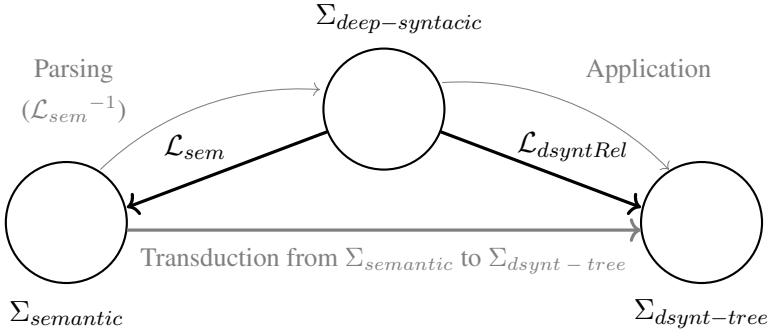


Figure 4: Illustration of application, parsing and transduction operations on Cousin (2023a)'s interface for semantic to deep-syntax transduction

ACG encoding of MTT We focus on Cousin (2023a)'s transduction between semantic and deep-syntax (see figure 4) and illustrate how we extend it.

$\Sigma_{\textit{semantic}}$ contains constants to represent the predicative semantic graphs (SemS), i.e., mostly semantemes and semantic relations. Constants of $\Sigma_{\textit{dsynt-tree}}$ are the ones to represent MTT's DSyntS, i.e., mostly lexemes, deep-syntactic relations and lexical functions. $\Sigma_{\textit{deep-syntactic}}$ is the abstract signature used to implement the

transduction between $\Sigma_{semantic}$ and $\Sigma_{dsynt\text{-}tree}$. Figure 5 gives the interpretation of some constants of $\Sigma_{deep\text{-}syntaciic}$ in $\mathcal{L}_{sem}$[1].

$\mathcal{L}_{sem}$

$$
\begin{aligned}
G, G' &:= n \Rightarrow t \\
charlie^{ds} &: G \\
&:= \lambda\, e0.\ (charlie^{s}\ e0) \\
taylor^{ds} &: G \\
&:= \lambda\, e0.\ (taylor^{s}\ e0) \\
invite^{ds} &: G' \rightarrow G \rightarrow G \\
&:= \lambda^{0}\ A\ x\ y.\ \lambda\, e_0.\ (\exists\, e_x . \exists\, e_y.\ (invite^{s}\ e_0) \\
&\wedge (R_1\ e_0\ e_x) \wedge (R_2\ e_0\ e_y) \wedge A \wedge (x\ e_x) \wedge (y\ e_y)) \\
son^{ds} &: G' \rightarrow G \rightarrow G \\
&:= \lambda^{0}\ x\ y.\ \lambda\, e_0.\ (\exists\, e_x.\ \exists\, e_y.\ (son^{s}\ e_0) \wedge (R_1\ e_0\ e_x) \\
&\wedge (R_2\ e_0\ e_y)\ \wedge (x\ e_x) \wedge (y\ e_y))
\end{aligned}
$$

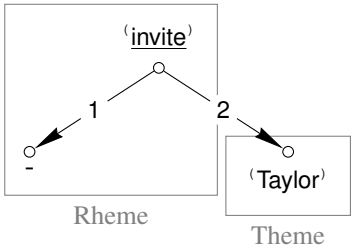
$\Sigma_{semantic}$

$$
\begin{aligned}
charlie^{s} &: n \Rightarrow t \\
taylor^{s} &: n \Rightarrow t \\
invite^{s} &: (n \Rightarrow t) \\
son^{s} &: (n \Rightarrow t)
\end{aligned}
$$

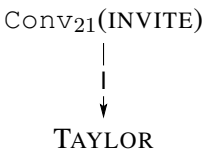
Figure 5: Interpretation of some constants of $\Sigma_{deep\text{-}syntactic}$ in $\mathcal{L}_{sem}$, and expression of their types in $\Sigma_{semantic}$. *(Notation: exponent -^{ds} is used to indicate that a constant belong to $\Sigma_{deep\text{-}syntacic}$ while -^{s} is used to indicate that a constant belong to $\Sigma_{semantic}$. Type G' is for optionally expressible semantic arguments. The difference with type G is not relevant for the purpose of this paper, and both can be treated as the same.)*

Hence, by parsing of $\mathcal{L}_{sem}$ and application of $\mathcal{L}_{dsyntRel}$, the DSyntS associated with a SemS is obtained. This requires that the SemS has an antecedent by $\mathcal{L}_{sem}$.

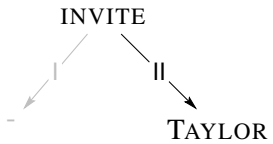
Cousin (2023a)'s encoding did not enable thematicity representation. It is especially problematic when a choice between the use of a copula or a noun phrase is needed (see figure 2 above with ⟨son⟩), or in the case of unexpressed optional semantic arguments, like in "*Taylor is invited*". The first actant of ⟨invite⟩ is not expressed (see figure 6). Its corresponding DSyntR (see figure 6b) only has one syntactic actant, and is different from the one obtained by Cousin (2023a) (see figure 6c).



(a) SemS and Sem-CommS associated with "*Taylor is invited.*"



(b) DSyntS associated with "*Taylor is invited.*" *(Conv$_{21}$: see (Mel'Čuk and Polguère 2021))*



(c) DSyntS associated with "*Taylor is invited.*" in Cousin (2023a)'s encoding

Figure 6: SemS and DSyntS of "*Taylor is invited*"

We propose to extend Cousin (2023a)'s encoding by adding type information corresponding to the communicative structure. Figure 7 shows the new hierarchy of ACGs with contants being typed according to their

[1]We use the same notations as Cousin (2023a)

thematicity in $\Sigma_{\textit{deep-syntactic-tr}}$. We use composition capabilities of ACGs to have this information percolate to $\Sigma_{\textit{semantic}}$ and $\Sigma_{\textit{dsynt-tree}}$, that does not need to change (see figure 8).

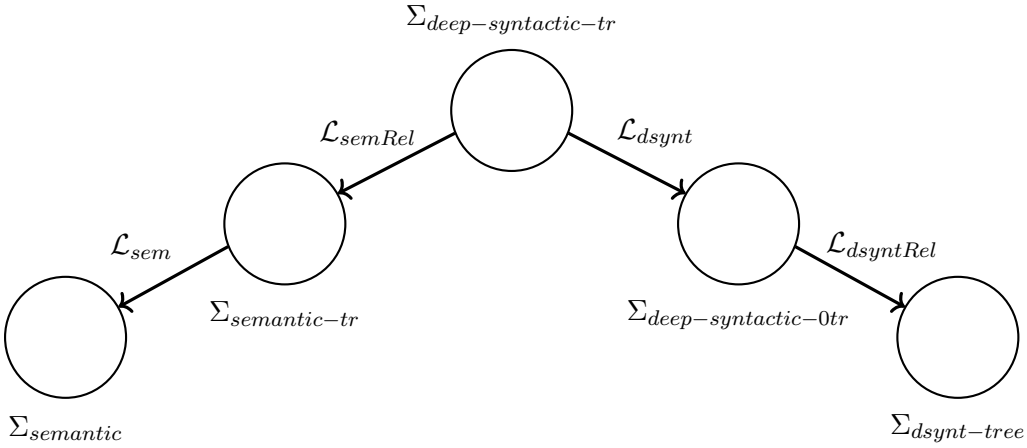


Figure 7: Hierarchy of the syntax-semantic interface of the implementation of MTT within ACGs that takes theme and rheme into account

Hence, by controlling the types of constants, $\Sigma_{\textit{deep-syntactic-tr}}$ controls how structures combine with respect of their thematic markers. Consequently, if a SemS having a theme-rheme division that does not make sense is parsed, no antecedent will be found. For instance, figure 8 shows that two abstract constants exist for ⟨son⟩, son^{dst}_{rtr} and son^{dst}_{rrr}. None of them allow ⟨son⟩ to have both its arguments typed as theme. The first case is the one of figure 2a, with son^{dst}_{rtr} where the first argument is theme and the second rheme (as well as ⟨son⟩ itself) which lead to the copula expression "*X is Y's son*". The second case is the one of figure 2b, with son^{dst}_{rrr}, where all constants are rheme, leading to the nominal expression "*X, Y's son*".

An intermediary deep-syntactic signature, $\Sigma_{\textit{deep-syntactic-0tr}}$, was also added. It erases all thematic makers present in $\Sigma_{\textit{deep-syntactic-tr}}$. The choice was made for now to focus on encoding thematicty in semantic but not to keep it in deep-syntax, since the role it plays there is less important than in the syntax-semantic interface.

Conclusion This paper presented how we added the theme-rheme opposition of the Sem-CommS to Cousin (2023a)'s encoding of MMT into ACG. This addition adds control over the structures by composing ACGs that bear thematicity information in their constant's typing. It enables the differentiation of SemR sharing the same SemS but not the same Sem-CommS, and hence the generation of different DSyntS associated to these SemR.

References

Cousin, Marie (June 2023a). "Meaning-Text Theory within Abstract Categorial Grammars: Towards Paraphrase and Lexical Function Modeling for Text Generation". In: *IWCS 2023 - 15th International Conference on Computational Semantics*. Ed. by Maxime Amblard and Helen Breitholtz. Nancy, France: Association for Computational Linguistics. Open archive HAL: hal-04104453. URL: https://inria.hal.science/hal-04104453.

Cousin, Marie (June 2023b). "Vers une implémentation de la théorie sens-texte avec les grammaires catégorielles abstraites". In: *18e Conférence en Recherche d'Information et Applications – 16e Rencontres Jeunes Chercheurs en RI – 30e Conférence sur le Traitement Automatique des Langues Naturelles – 25e Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues*. Ed. by Marie Candito, Thomas Gerald, and José G Moreno. Paris, France: ATALA, pp. 72–86. Open archive HAL: hal-04100197. URL: https://inria.hal.science/hal-04100197.

de Groote, Philippe (July 2001). "Towards Abstract Categorial Grammars". In: *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*. Toulouse, France: Association for Computational Linguistics, pp. 252–259. DOI: 10.3115/1073012.1073045. ACL anthology: P01-1033.

Kanazawa, Makoto (June 2007). "Parsing and Generation as Datalog Queries". In: *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Prague, Czech Republic: Association for Computational Linguistics, pp. 176–183. ACL anthology: P07-1023.

$\mathcal{L}_{semRel}$

$$
\begin{aligned}
G, G' &:= n_t \Rightarrow t \\
G_t, G'_t &:= n_t \Rightarrow t \\
G_r, G'_r &:= n_r \Rightarrow t \\
C_{t\to r} &: G_t \to G_r \\
C_{r\to\emptyset} &: G_r \to G \\
C_{t\to\emptyset} &: G_t \to G \\
charlie^{dst} &: G \\
&:= \lambda\, e0.\ (charlie^{str}\ e0) \\
taylor^{dst} &: G \\
&:= \lambda\, e0.\ (taylor^{str}\ e0) \\
invite^{dst}_{rtr} &: G'_t \to G_r \to G_r \\
&:= \lambda^0\, A.\ \lambda^0\, x\ y.\ \lambda\, e_0.\ (\exists\, e_x.\exists\, e_y.\ (invite^{str}\ e_0) \\
&\wedge (R_1\ e_0\ e_x) \wedge (R_2\ e_0\ e_y) \wedge A \wedge (x\ e_x) \wedge (y\ e_y)) \\
invite^{dst}_{rrt} &: G'_r \to G_t \to G_r \\
&:= \lambda^0\, A.\ \lambda^0\, x\ y.\ \lambda\, e_0.\ (\exists\, e_x.\exists\, e_y.\ (invite^{str}\ e_0) \\
&\wedge (R_1\ e_0\ e_x) \wedge (R_2\ e_0\ e_y) \wedge A \wedge (x\ e_x) \wedge (y\ e_y)) \\
son^{dst}_{rtr} &: G'_t \to G_r \to G_r \\
&:= \lambda^0\, x\ y.\ \lambda\, e_0.\ (\exists\, e_x.\ \exists\, e_y.\ (son^{str}\ e_0) \wedge (R_1\ e_0\ e_x) \\
&\wedge (R_2\ e_0\ e_y)\ land(x\ e_x) \wedge (y\ e_y)) \\
son^{dst}_{rrr} &: G'_r \to G_r \to G_r \\
&:= \lambda^0\, x\ y.\ \lambda\, e_0.\ (\exists\, e_x.\ \exists\, e_y.\ (son^{str}\ e_0) \wedge (R_1\ e_0\ e_x) \\
&\wedge (R_2\ e_0\ e_y)\ land(x\ e_x) \wedge (y\ e_y))
\end{aligned}
$$

$\Sigma_{semantic\text{-}tr}$

$$
\begin{aligned}
charlie^{str} &: n \Rightarrow t \\
taylor^{str} &: n \Rightarrow t \\
invite^{str}_{rtr} &: (n_t \Rightarrow t) \to (n_r \Rightarrow t) \to (n_r \Rightarrow t) \\
invite^{str}_{rtr} &: (n_r \Rightarrow t) \to (n_t \Rightarrow t) \to (n_r \Rightarrow t) \\
son^{str}_{rtr} &: (n_t \Rightarrow t) \to (n_r \Rightarrow t) \to (n_r \Rightarrow t) \\
son^{str}_{rrr} &: (n_r \Rightarrow t) \to (n_r \Rightarrow t) \to (n_r \Rightarrow t)
\end{aligned}
$$

Figure 8: Interpretation of some constants of $\Sigma_{deep\text{-}syntactic\text{-}tr}$ in $\mathcal{L}_{semRel}$ and expression of their types in and $\Sigma_{semantic\text{-}tr}$ *(Notation: exponent -^{dst} is used to indicate that a constant belong to $\Sigma_{deep\text{-}syntacic\text{-}tr}$ while -^{str} is used to indicate that a constant belong to $\Sigma_{semantic\text{-}tr}$. Type G' is for optionally expressible semantic arguments. The difference with type G is not relevant for the purpose of this paper, and both can be treated as the same.)*

Mel'čuk, I.A. (2012). *Semantics: From Meaning to Text*. Vol. 1. Semantics: From Meaning to Text. John Benjamins Publishing Company.

Mel'čuk, I.A. (2013). *Semantics: From Meaning to Text*. Vol. 2. Semantics: From Meaning to Text. John Benjamins Publishing Company.

Mel'čuk, I.A. (2015). *Semantics: From Meaning to Text*. Vol. 3. Semantics: From Meaning to Text. John Benjamins Publishing Company.

Mel'Čuk, Igor and Alain Polguère (July 2021). "Les fonctions lexicales dernier cri". In: *La Théorie Sens-Texte. Concepts-clés et applications*. Ed. by Sébastien Marengo. Dixit Grammatica. L'Harmattan, pp. 75–155. Open archive HAL: hal-03311348.

Polguère, Alain (1990). "Structuration et mise en jeu procédurale d'un modèle linguistique déclaratif dans un cadre de génération de texte". PhD thesis. URL: https://olst.ling.umontreal.ca/static/pdf/PolguerePhD1990.pdf.