

Building argumentative adpositional trees: Towards a high-precision method for reconstructing arguments in natural language

FEDERICO GOBBO & JEAN H.M. WAGEMANS

*ACLCL – Amsterdam Centre for Language and Communication / StudiUm – Studi Umanistici
University of Amsterdam / University of Turin
The Netherlands / Italy
f.gobbo@uva.nl / federico.gobbo@unito.it*

*ACLCL – Amsterdam Centre for Language and Communication
University of Amsterdam
The Netherlands
j.h.m.wagemans@uva.nl*

ABSTRACT: There is a need for a tool for reconstructing arguments that describes their linguistic elements with high precision and at the same time identifies their type. In this paper, we prepare the ground for developing such a tool by introducing the notion of ‘argumentative adpositional tree’. The notion is based on a combination of the linguistic representation framework of *Constructive Adpositional Grammars* (CxAdGrams) and the argument classification framework of the *Periodic Table of Arguments* (PTA).

KEYWORDS: argumentation, argumentative adpositional trees, Constructive Adpositional Grammars, formal linguistics, natural language processing, Periodic Table of Arguments, reconstructing arguments

1. INTRODUCTION

An argumentative analysis usually proceeds from the original text in natural language to a reconstructed version that highlights its argumentative aspects. In order to grasp, for example, what is at stake in Donald J. Trump’s tweet pictured in Figure 1, an analyst may have to reconstruct the original text as *I am the world’s greatest writer of 140 character sentences, because many people have said so.*¹



Figure 1 The original text of a tweet by Donald J. Trump

During the reconstruction process, the analyst has to address a number of important issues. One of them concerns the selection of those parts of the original text that are relevant to include in the reconstruction. Whereas a minimal reconstruction includes only the statements that function as premise and conclusion, more sophisticated ones may also include statements expressing

¹ For a more detailed analysis of this example see Wagemans (2017, URL = www.periodic-table-of-arguments.org/periodic-table-of-arguments/delta-quadrant/argumentum-ad-populum)

doubt or criticism with regard to the acceptability of a certain point of view, statements expressing common starting points, and statements that relate to procedural aspects of an argumentative discussion such as the division of the burden of proof.²

A subsequent issue the analyst has to address is how to represent the selected parts of the original text in the reconstruction. The decisions regarding this issue are usually taken on the basis of the envisioned aim of the analysis. When that aim is, for instance, to obtain an overview of the all the premises that the arguer has put forward in support of the conclusion, it may suffice to provide a numbered list of literal representations of the statements that function as such. But if the aim is to provide an evaluation of the quality of the individual arguments, the analyst may want to represent the original statements in such a way that it becomes possible to determine the role the individual linguistic elements of each of the premises play in supporting the conclusion. This is a challenging task, even more since there is so far no analytical tool for reconstructing arguments that describes their linguistic elements in great detail and at the same time identifies their type.

In this paper we present a new method for reconstructing arguments in natural language that enables the analyst to perform this task. Our method centers around the notion of ‘argumentative adpositional tree’ (or ‘argumentative adtree’). Like its linguistic counterpart, an argumentative adtree represents sentences on the morphosyntactic level. In addition, it contains pragmatic information regarding the argumentative function of their constituents and the type of argument they instantiate.

The notion of ‘argumentative adtree’ is the result of combining two theoretical frameworks. Its basic characteristics are derived from *Constructive Adpositional Grammars* (CxAdGrams), a linguistic representation framework developed by Gobbo and Benini (2011) that employs adtrees for the purpose of representing natural language. The addition of a layer of pragmatic information to these adtrees is carried out by using the *Periodic Table of Arguments* (PTA), an argument classification framework developed by Wagemans (2016) that is especially suitable for formal linguistic and computational approaches to argument.

We begin the paper with a general introduction to the two frameworks involved. In Section 2, we lay out the fundamentals of CxAdGrams. We explain the theoretical starting points of this approach as well as the central notion of ‘adtree’. In Section 3, we discuss the way in which the PTA describes and classifies the types of argument. We describe the three partial characteristics of argument that constitute its theoretical framework and provide two concrete examples of so-called ‘first-order arguments’. Then, in Section 4, we combine the two frameworks and introduce the notion of ‘argumentative adtree’. We illustrate its use by generating and elucidating the argumentative adtrees of the two examples presented in the previous section. Finally, in Section 5, we briefly summarize and discuss our method of reconstructing arguments in natural language and indicate the main directions for further research.

2. CONSTRUCTIVE ADPOSITIONAL GRAMMARS

The theoretical framework of *Constructive Adpositional Grammars* (CxAdGrams) is the result of the application of constructive mathematics to the adpositional paradigm in linguistics. We first elucidate this framework by explaining the meaning of the key terms.

The adpositional paradigm in linguistics is the idea that each pair of linguistic elements can be conveniently described in terms of asymmetrical relations, that is, in such a way that their arrangement cannot be reversed. Thus, given a pair of morphemes, words or expressions,

² For an overview of analytically relevant contributions to an argumentative discussion, see van Eemeren and Grootendorst (2004, p. 68).

there is always one element that ‘governs’ the other, and consequently, the latter element ‘depends’ on the former. An example is the phrase *Children play*, which has the verb *play* as the governing element (*gov*) and the noun *children* as the dependent element (*dep*). The hierarchical relation between a pair of linguistic terms is conventionally called an ‘adposition’ and can be pictured in a so-called ‘adpositional tree’ (or ‘adtree’).

The set of rules for building adtrees that is admissible within a given natural language forms an ‘adpositional grammar’ (or ‘adgram’). The term ‘adgrams’ denotes all the possible grammars of any human language. In CxAdGrams, more specifically, the formation of adgrams follows certain meta-rules that are described in terms of Grothendieck topoi – the most important instrument in topos theory, a branch of constructive mathematics. Since in constructive mathematics, unlike in classical logic, the use of the so-called ‘law of excluded middle’ is not allowed, the information content of any statement regarding the formulas of a theorem is strictly preserved – see Bridges and Richman (1987).

There is a tradition of using constructive mathematics to formally represent natural languages, starting from the work of Adjukiewicz (1935) and Church (1940). CxAdGrams, to the best of our knowledge, is the only framework that uses topos theory for this purpose. As a result, the adtrees it produces do not only represent natural language expressions in the form of recursive trees but can also be interpreted as formulas – which means that they are suitable for the purpose of natural language processing.³

We now turn to explaining the notion of ‘adpositional tree’ in more detail. A minimal adtree consists of a pair of linguistic elements and their relation, expressed in terms of their adposition. Figure 2 shows the abstract structure of such a minimal adtree – adapted from Gobbo and Benini (2011, p. 15).

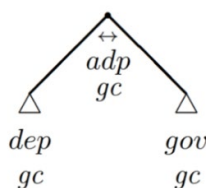


Figure 2 The abstract structure of a minimal adtree

Within this adtree, the positions of the governing and the dependent elements are conventional: the governor (*gov*) is put on the right leaf at the bottom of the rightmost branch, while the dependents (*dep*) are put on the left leaves at the bottom of the branches on the left (in this case, there is only one). Their relation is represented by the adposition (*adp*), depicted as a hook under the bifurcation of the two branches. The variable *gc* under the hook and the leaves means ‘grammar character’. Finally, the triangles on the leaves (Δ) indicate that adtrees can be applied recursively one under the other, if needed.

While each dependent can have one and only one governor, a governor can rule more than one dependent. The exact number of dependents, which ultimately determines the shape of the adtree, is defined by the Tesnerian concept of valency.⁴ In order to illustrate this concept, we picture in Figure 3 the adtree of the phrase *Children play* that was mentioned above.

³ The linguistic and formal rules of CxAdGrams are not discussed here for reasons of conciseness. For a comprehensive presentation of this approach to linguistic analysis, see Gobbo and Benini (2011). The formal model is presented in Appendix B of this work.

⁴ The concept of valency was introduced by Tesnière (1959, 2015) within the framework of Structural Syntax. Gobbo and Benini (2013) clarify the relation between that framework and CxAdGrams.

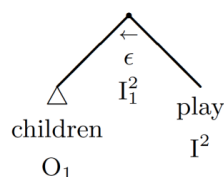


Figure 3 The adtree of the phrase *children play*

The adposition (*adp*) is in this case instantiated by an epsilon (ϵ), indicating that there is a syntactic relation between the two words. The triangle (Δ) indicates an adtree that represents morphological information regarding the word *children* and is hidden because it is irrelevant for our purposes. The grammar characters (*gc*) are in this case instantiated by O, I, and I. The theoretical framework of CxAdGrams uses five different grammar characters, which are represented by five vowels (A, E, I, O, U). Table 1 explains their meaning – adapted from Gobbo and Benini (2011, p. 41).

<i>gc</i>	<i>name</i>	<i>function</i>	<i>examples</i>
A	adjunctive	modifier of O	adjectives, articles, determiners
E	circumstantial	modifier of I	adverbs, adverbial expressions
I	verbant	valency ruler	verbs, interjections
O	stative	actants	nouns, pronouns, name-entities
U	underspecified	transferer	prepositions, derivational morphemes

Table 1 The meaning of grammar characters in adtrees

The shape of an adtree is mainly defined by verbants – typically, verbs (see Table 1). Their grammar characters (I) and those of the correlated nominal expressions (O) may show additional parameters. In the case of verbants, an apex indicates the verbal valency (*val*), i.e., the number of actants that are potentially involved in the activity described by the verb. The I^2 in our example indicates that the verb *play* is bivalent ($val = 2$), as semantically it implies a player (the first actant) as well as a game or a musical instrument (the second actant). In the case of the nominal expressions correlated with the verbant, a pedix indicates the number by which they are identified. In our example, *children* acts as the first actant (O_1), while the second actant has remained implicit. Finally, the information expressed in the complete adtree is summarized by the grammar character I_1^2 under the hook. Here, again, the apex indicates the valency value (*val*) of the verb, while the pedix indicates the number of actants (*act*) present in the sentence.⁵ Since the former is bigger than the latter ($val = 2$ and $act = 1$), the verb is only partially saturated.

We now discussed most of the basic aspects of linguistic adtrees. The meaning of the arrows – for instance the left arrow (\leftarrow) above the epsilon in Figure 3 and the generic leftright arrow (\leftrightarrow) above *adp* in Figure 2 – is explained in Section 4.

3. PERIODIC TABLE OF ARGUMENTS

The *Periodic Table of Arguments* (PTA) is a classification of argument that integrates the traditional dialectical accounts of argument schemes and fallacies as well as the rhetorical accounts of logical, ethotic, and pathetic means of persuasion into a systematic and

⁵ Please note that if the adposition of the second actant of *to play* is filled by the preposition *with*, the overall semantics slightly changes. Thus, in CxAdGrams, *to play* and *to play with* are considered different verbs.

comprehensive whole.⁶ The theoretical framework of the table is based on three partial characterizations of argument, namely (1) as first-order or second-order arguments; (2) as predicate or subject arguments; and (3) as a specific combination of types of statements. The superposition of these three partial characterizations yields a factorial typology of argument that can be used in order to develop tools for analyzing, evaluating, and producing argumentative discourse.

The types of arguments described in the PTA are ‘atomic’ in the sense that they consist of exactly one premise and one conclusion, both of which are expressed by means of a statement that consists of a subject and a predicate. Closely following logical conventions, subjects are indicated with letters *a*, *b*, etc., predicates with letters *X*, *Y*, etc. (predicate \top having the fixed meaning ‘true’), and complete propositions with letters *p*, *q*, etc.

The classification of the types of argument takes place by determining the ‘argument form’, a notion that comprises the first two partial characteristics mentioned above, and by subsequently determining the combination of types of statements instantiated by the argument, which yields the third partial characteristic. Without going too much into the details of the Argument Identification Procedure (Wagemans, 2018), we turn to mentioning the most important ingredients of the argument classification framework of the PTA.

From the above description it follows that the determination of the argument form involves an identification of the argument as either a first-order or a second-order argument and as either a predicate or a subject argument. These distinctions allow for four different possibilities: first-order predicate arguments, first-order subject arguments, second-order subject arguments, and second-order predicate arguments. For this reason, the visual representation of the PTA consists of a plane that is divided into four quadrants (Wagemans, 2017). The argument forms just mentioned correspond to the quadrants α , β , γ , and δ respectively. In Table 2, for each quadrant we list the corresponding argument form and provide a concrete example.

<i>quadrant</i>	<i>argument form</i>	<i>example</i>
α	<i>a is X, because a is Y</i>	The subject (<i>a</i>) is driving fast (<i>X</i>), because he (<i>a</i>) left a long trace of rubber on the road (<i>Y</i>)
β	<i>a is X, because b is X</i>	Cycling on the grass (<i>a</i>) is forbidden (<i>X</i>), because walking on the grass (<i>b</i>) is forbidden (<i>X</i>)
γ	<i>q is \top, because r is \top</i>	He must have gone to the pub (<i>q</i>), because the interview was cancelled (<i>r</i>)
δ	<i>q is \top, because q is Z</i>	We only use 10% of our brain (<i>q</i>), because that (<i>q</i>) was said by Einstein (<i>Z</i>)

Table 2 Argument forms and examples in the four quadrants of the PTA

The argument types situated within each of the quadrants are further differentiated on the basis of a determination of the specific combination of types of statements they instantiate. For this purpose, the PTA makes use of a tripartite typology consisting of statements of fact (*F*), statements of value (*V*), and statements of policy (*P*). The conclusion and premise of the argument may thus instantiate one of the following nine combinations of types of statements: *PP*, *PV*, *PF*, *VP*, *VV*, *VF*, *FP*, *FV*, *FF*. An example is *The government should invest in jobs, because this will lead to economic growth*, which has a statement of policy (*P*) as its conclusion and a statement of fact (*F*) as its premise and therefore instantiates the combination ‘*PF*’.

⁶ The present explanation of the theoretical framework of the *Periodic Table of Arguments* is based on Wagemans (2016, 2017, 2018).

When taken together, the three partial characterizations of argument constitute a theoretical framework that allows for $2 \times 2 \times 9 = 36$ systematic types of arguments. Each of these systematic types hosts a number of ‘isotopes’, which are named in accordance with the existing dialectical and rhetorical classifications of argument classification. The traditional names usually originate in the linguistic formulation of the relation between the premise and the conclusion. The argument *The suspect was driving fast, because he left a long trace of rubber on the road*, for instance, can be identified as a first-order predicate argument that combines a statement of fact with another statement of fact. The systematic name of this argument is therefore ‘1 pre FF’. Given that the relation between the premise and the conclusion can be captured by saying that the predicate of the statement expressed in the premise, *leaving a long trace of rubber on the road*, is an ‘effect’ of the predicate of the conclusion, *driving fast*, the traditional name of this specific isotope of ‘1 pre FF’ is ‘argument from effect’.

Within every quadrant, the systematic place of the type of argument is determined by the specific combination of types of statements it instantiates (*FF*, *VF*, *PF*, etc.), while the isotopes representing the traditional names are placed in a vertical line. In Figure 4, we picture the current version of the PTA – for updates and more detailed analyses of examples, see its official website www.periodic-table-of-arguments.org.

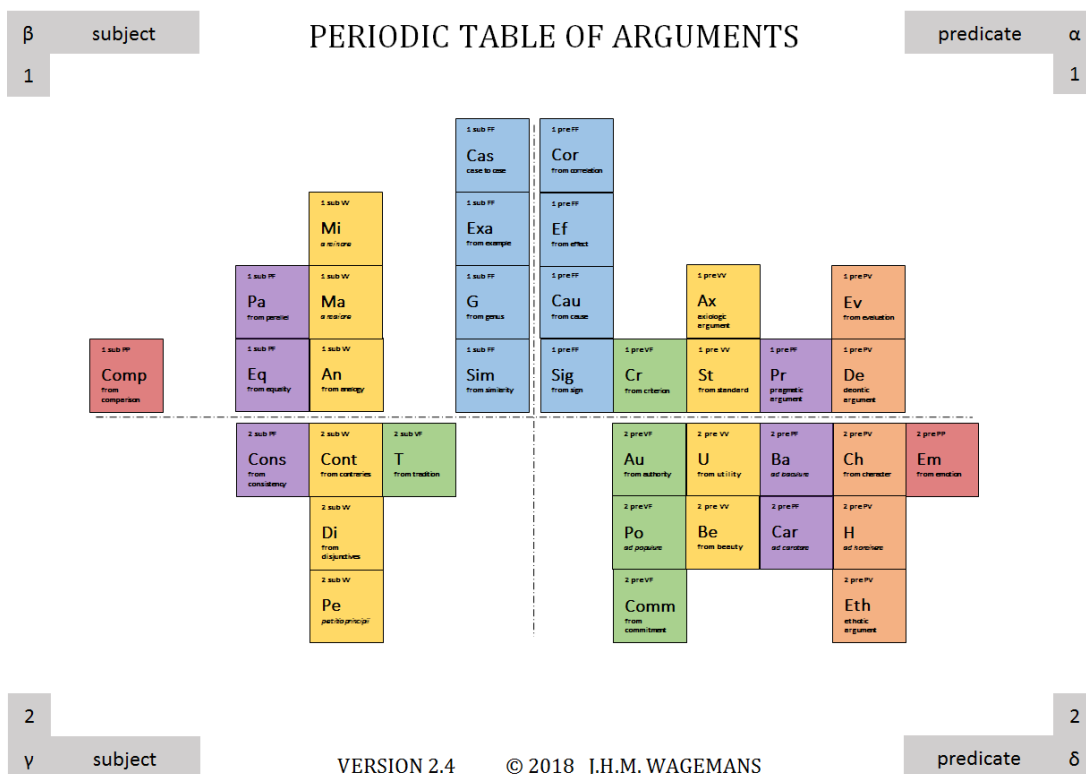


Figure 4 Version 2.4 of the *Periodic Table of Arguments*

In the next section we demonstrate how the theoretical framework of the PTA can be used for enriching the linguistic adtrees generated by CxAdGrams with pragmatic information regarding the type of argument. We do so by providing an analysis of two of the concrete examples of arguments that were mentioned in Table 2. The first example, *The suspect (a) was driving fast (X), because he (a) left a long trace of rubber on the road (Y)*, is a first-order predicate argument

that combines a statement of fact with another statement of fact ('1 pre FF') and is traditionally known as the 'argument from effect'. The second example, *Cycling on the grass (a) is forbidden (X), because walking on the grass (b) is forbidden (X)*, is a first-order subject argument that combines a statement of value with another statement of value ('1 sub VV') and is traditionally known as the 'argument from analogy'.

4. COMBINING CXADGRAMS AND PTA

Although CxAdGrams is built mainly for expressing morphology and syntax, the adtrees generated by its theoretical framework are also suitable for expressing pragmatics.⁷ In this section we show how to insert pragmatic information about the type of argument into a linguistic adtree, thereby developing the notion of 'argumentative adtree'.⁸

As explained in the previous section, the PTA takes an argument to be formed by two statements, each of which consist of a subject (indicated by *a, b*, etc.) and a predicate (indicated by *X, Y*, etc.). In the corresponding adtrees of these individual statements, the predicate is represented by the leaf of the rightmost branch. While the position of the predicate does not change in transforming linguistic into argumentative adtrees, in the latter the subject is emphasized. In particular, the first actant (O_1), which corresponds to the subject, is put in evidence as the topmost left branch of the tree. The remaining linguistic material automatically becomes part of the predicate.

For the analyst, the identification of the subject and the predicate of the statements involved in the argument can be complicated by the fact that the arguer has left certain linguistic elements implicit. This is a well-known problem in argumentation analysis, for which our method provides an extra tool. In order to represent the elements that function in the argumentation in an adequate way, the analyst can consider the valency of the verb which rules the predicate and check whether or not the verb is saturated. If in the original text some actants are explicitly stated and some others have remained implicit, the implicit actants should be inferred and extracted from the semantic interpretation of the statement itself. This procedure of elicitation of the in-valent structure eventually deepens the analysis of the argument in terms of robustness. In fact, the overall goal of the reconstruction is to understand better how the argument works, and the analysis of the in-valent structure can play a major role in achieving this goal.

After having identified the subject and the structure of the predicate of the two statements, the analyst can determine their argumentative function and collocate the argument in the PTA. In the argumentative adtree, the conclusion is indicated by σ (sigma), standing for the Greek equivalent *συμπέρασμα* (*sumpérasma*), and the premise by π (pi), standing for *πρότασις* (*prótasis*). Regarding the order of presentation of these statements in the original text, van Eemeren and Snoeck Henkemans (2016, p. 33) distinguish between a progressive and a retrogressive mode. The former presents first the premise (π) and then the conclusion (σ), connecting them by means of the conjunction *so, therefore*, or another one with a similar function. Since there is a multitude of equivalent expressions, in argumentative adtrees of reconstructed arguments the progressive mode is represented only formally, by a left arrow (\leftarrow). The retrogressive mode starts from the conclusion (σ) and then arrives at the premise (π), connecting them with a conjunction such as *because, since*, etc. This mode is represented in argumentative adtrees by a right arrow (\rightarrow). For the sake of simplicity and conciseness, the

⁷ Gobbo and Benini (2011, Ch. 6) have already provided pragmatic adtrees, representing the relation between illocutionary and locutionary acts as described in Searle (2010).

⁸ The transformation from linguistic adtrees to argumentative adtrees is formally justified by the so-called conjugate construction in the formal model of *CxAdGrams* – see Gobbo and Benini (2011).

examples we have chosen for our analysis below are retrogressive in nature and use *because* as a conjunction. In the corresponding argumentative adtrees, this is represented with a right arrow (\rightarrow).

Figure 5 shows the argumentative adtree of the two main types of first-order arguments distinguished in the PTA in an abstract way, i.e. showing their regressive normal form, regardless of the linguistic information. In each argumentative adtree, the adpositions representing the premise (π) and the conclusion (σ) appear once and once only. On the left we pictured the adtree of first-order predicate arguments, situated in the alpha quadrant (α) and having the form “*a is X, because a is Y*”, and on the right the adtree of first-order subject arguments, situated in the beta quadrant (β) and having the form “*a is X, because b is X*”.

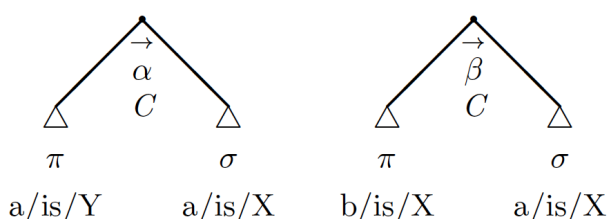


Figure 5 The abstract argumentative adtrees of first-order arguments

In both adtrees, the relation between the premise (π) and the conclusion (σ) is represented by the adposition, which is the hook. The quadrant is indicated under the arrow. The letter *C* indicates generically the combination of the types of statements (*FF*, *VF*, *PF*, etc.) instantiated by the argument. The quadrant and combination indicators together enable the analyst to identify which of the 36 possible types of argument distinguished in the PTA describes the concrete argument under scrutiny.

We now turn to illustrating our method of reconstructing arguments by building the argumentative adtrees of the two concrete examples mentioned at the end of the previous section. Both are examples of first-order arguments, their variation lies in the other two partial characteristics that constitute the theoretical framework of the PTA. The first example is a predicate argument that supports a statement of fact with another statement of fact (‘1 pre FF’) and the second a subject argument that supports a statement of value with another statement of value (‘1 sub VV’).

Example 1

The first example we analyze is *The suspect was driving fast, because he left a long trace of rubber on the road*. Figures 6 and 7 picture the linguistic adtrees respectively of the premise and the conclusion of the example.⁹ At first, a trivial parser will identify the two verbal forms that rule the two phrases: respectively, *was driving* (ruling the conclusion, which is the governor of the sentence and is therefore put at the rightmost subtree) and *left* (ruling the premise, which depends on the conclusion). At this point, the respective first actants (O_1) are identified, i.e. *the suspect* and *he*. In the conclusion, there is an adverb, *fast*, that specifies a quality of the verb (therefore, it has the grammar character E). In the premise, a similar role is played by the locative expression *on the road*. Its global adposition is E, and its internal structure contains a preposition (U), a noun (O) and a determiner (A). Unlike in the conclusion, in the premise there is a complex expression, *a long trace of rubber*, that saturates the second valency (O_2). Finally,

⁹ Conclusion and premise are connected by the conjunction *because*. For the reader’s sake, the two adtrees are presented separately, so the conjunction is not represented in Figures 6 and 7. Please note that in some adtrees, some branches are longer than others. This happens just for human readability; when linearized for machine coding, all branches become equally long.

the triangles Δ are used to hide linguistic details that are not immediately relevant for the present purpose – in this case, the morphological information regarding *was driving*.

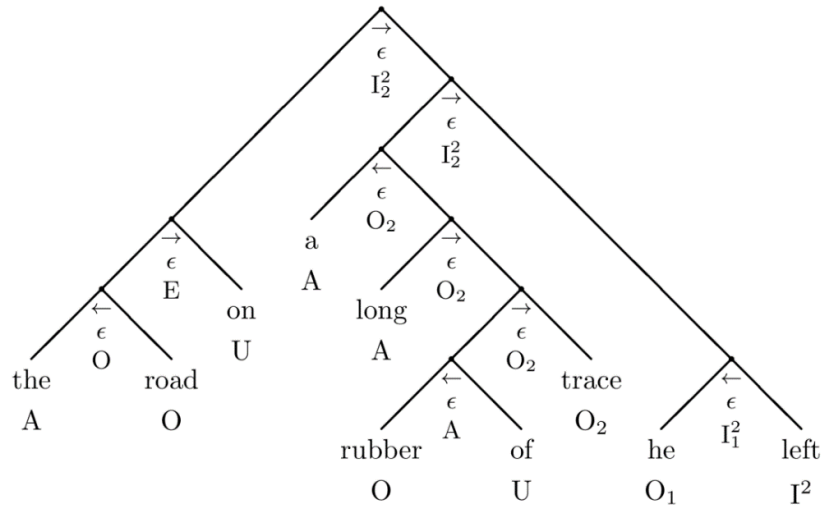


Figure 6 The fully expanded linguistic adtree of the premise of Example 1

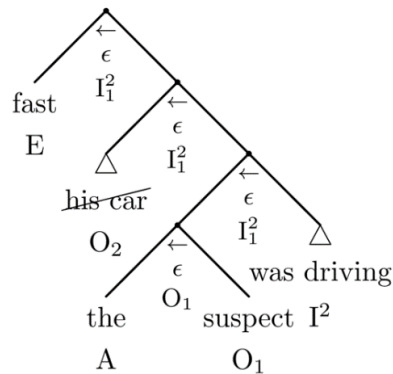


Figure 7 The linguistic adtree of the conclusion of Example 1

At this point, we can transform the linguistic adtree into the argumentative one. We take the premise *he left a long trace of rubber on the road* as an example and show how the transformations mentioned above take place (see Figure 8). The argumentative function of this sentence, a premise, is reflected in the adposition π , which is placed under the topmost hook. Then, the analyst identifies the type of statement expressed in the premise as a statement of fact. In the argumentative adtree, this information is put as grammar character F under adposition π .

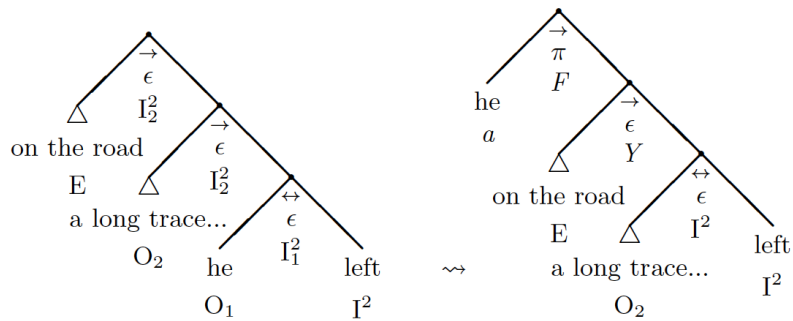


Figure 8 The transformation from linguistic to argumentative adtree

After having inserted this argumentative information in the adtree, the analyst proceeds with the extraction of the subject (*a*), which is put at the leftmost subtree, leaving the other elements of the sentence as parts of the predicate (*Y*). The extraction, which in this case results in a different position of the leaf *he*, highlights the subject-predicate structure of the sentence, which in turn enables the identification of the type of argument.

Then, after a similar procedure is carried out for the conclusion (σ), the analyst can put the two adtrees together (see Figure 9). This will create a new adtree with pragmatic information about the type of argument in the topmost hook. In this case, the premise and conclusion together form a first-order predicate argument (α) that combines a statement of fact with another statement of fact (*FF*).

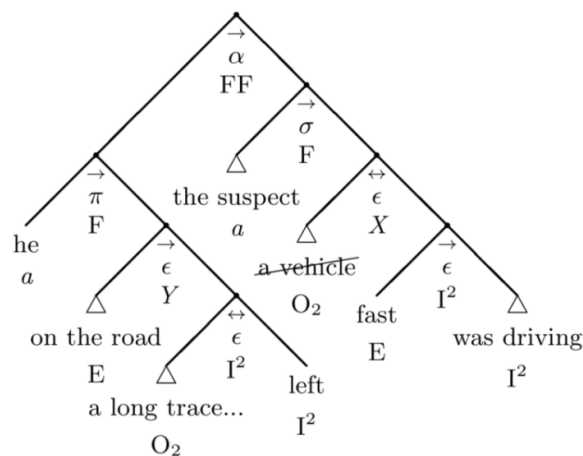


Figure 9 The argumentative adtree of Example 1

The analyst is helped in her work by the linguistic analysis performed through CxAdGrams. In fact, the in-valent structure reveals that the verbal form ruling the conclusion, *was driving*, implies the possible presence of two actants: the driver (O_1) and the vehicle (O_2). In this example, the second actant (O_2) is not expressed, which is why that information is hidden. An adtree can show hidden information under the form of a barred leaf. Although at first sight, this hidden information does not seem to be that relevant, on closer inspection it could be crucial for the argumentation analysis, especially when the analyst is dealing not with a single argument but with a whole argumentative text. In fact, if the analyst forgets to represent hidden information, she risks to lose an important piece of information. Omissions can be very important to understand if the argument structure of the whole text is robust or not, and in

particular to detect the weak points in the reconstruction. In this example, the subject, *the suspect (a)*, is metonymically identical with the driver (O_1) even if, strictly speaking, it is the vehicle (O_2) that *left the long trace on the road*, not the subject. This shows an important difference between linguistic adtrees and argumentative adtrees: what is peripheral from a linguistic point of view, can be central from an argumentative perspective. In particular, the circumstantial *fast (E)* of the premise (π) is linguistically merely a decoration of the ruling verbal form *was driving (I²)*. But it plays a central role in the persuasive force of the argument: if the suspect weren't driving *fast (E)*, he could have never *left (I²) a long trace on the road (O₂)*. In other words, *long* is the key term and it is related to *fast*.

Example 2

The second example is *Cycling on the grass is forbidden, because walking on the grass is forbidden*. Like with the previous example, the analyst should start from the linguistic analysis of the two statements that form the argument. Figure 10 shows the linguistic adtree, whose main subtrees, representing respectively the main and the secondary phrase, are almost identical, as they share the same linguistic structure. It is worth noting that the preposition *on* is the adposition of the two identical expressions *on the grass* that modify the subjects (O_1) of both subtrees, and therefore takes the adjunctive grammar character (A).

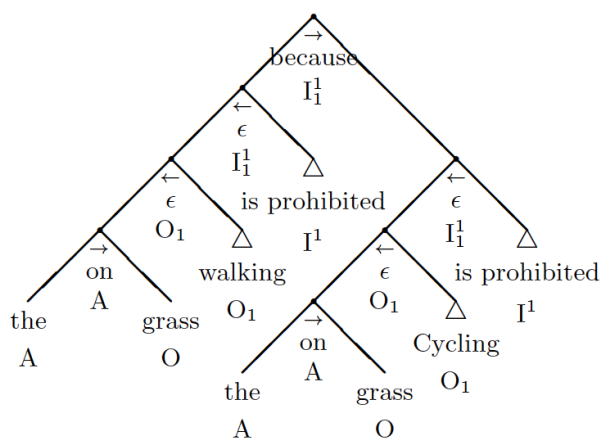


Figure 10 The linguistic adtree of Example 2

The argumentative counterpart of the linguistic adtree is quite similar – compare Figures 10 and 11. As both statements share the same predicate, *is prohibited (X)*, but have different subjects, *cycling on the grass (a)* and *walking on the grass (b)*, they instantiate a first-order subject argument. This information is inserted by putting a β , standing for the corresponding quadrant of the *Periodic Table of Arguments*, under the topmost hook. Finally, since their verbal form is a cue for labelling both phrases as statements of value, the analyst adds *VV* as an indication of the combination of types statements involved.

Our reconstruction reveals that the linguistically peripheral elements *on the grass* play a vital role in the argument and are therefore pragmatically central. In fact, if we cut them out, the resulting sentence of the argument *Cycling is prohibited because walking is prohibited*, loses all argumentative force.

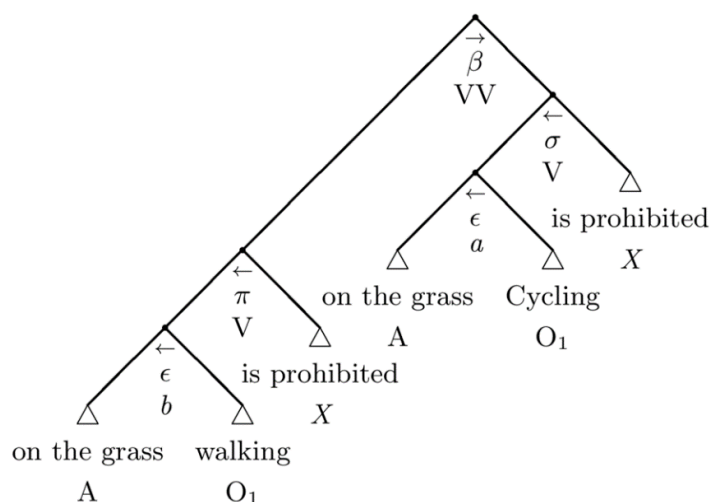


Figure 11 The argumentative adtree of Example 2

As illustrated by reconstructing these two examples, argumentative adtrees are powerful analytical tools that enable the analyst to show where the pragmatic force is placed within the linguistic material.

5. CONCLUSION

In this paper we demonstrated a high precision tool for the purpose of reconstructing arguments in natural language that centers around the notion of ‘argumentative adpositional tree’. The tool is the result of the combination of two theoretical frameworks, the linguistic representation framework of *Constructive Adpositional Grammars* (CxAdGrams) and the argument classification framework of the *Periodic Table of Arguments* (PTA).

Argumentative adpositional trees not only represent in great detail the linguistic features of premises and conclusions of the argument under scrutiny but they also contain pragmatic information about the type of argument. Moreover, the tool permits to express hidden information contained in the valency structure if this is necessary or desirable in light of the aim of the analysis.

We believe that our method of reconstructing arguments has some advantages compared to existing methods. In present-day argumentation theory, the analysis of the external organization of the argumentation – the so-called ‘argumentation structure’ – and the analysis of the internal organization of an argument – the so-called ‘argument scheme’ are often kept separated. Building argumentative adtrees enables the analyst to obtain an integrated picture of these two aspects of argumentative discourse. The method thus permits the analyst not only to operate on the level of the individual words, but also to freely choose the level of linguistic detail to be shown in the reconstruction in relation to the specific aim of the analysis.

A further direction of research is to apply the procedure for creating argumentative adtrees to the two main types of second-order arguments, which are situated in the gamma and delta quadrants of the PTA. Also, since arguments rarely appear in isolation, it is important to extend the procedure for generating argumentative adtrees so as to be able to represent complex concatenations of arguments and, ultimately, a complete argumentative text.

ACKNOWLEDGEMENTS: This paper is the result of joint work by the two authors. Federico Gobbo is mostly responsible for Section 2, Jean H.M. Wagemans for Section 3, whereas all other sections are rather the result of collaborative efforts. The authors thank Marco Benini for his thorough reading of the manuscript, and in particular for checking the parts concerning constructive mathematics.

REFERENCES

- Adjukiewicz, K. (1935). Die syntaxische Konnexität. *Studia Philosophica*, 1, 1-27.
- Bridges, D., & Richman, F. (1987). *Varieties of Constructive Mathematics*. Cambridge: Cambridge University Press.
- Church, A. (1940). A formulation of the simple theory of types. *Journal of Symbolic Logic*, 5, 56-58.
- Eemeren, F. H. van, & Grootendorst, R. (2004). *A systematic theory of argumentation. The pragma-dialectical approach*. Cambridge: Cambridge University Press.
- Eemeren, F. H. van, & Snoeck Henkemans, A. F. (2016). *Argumentation. Analysis and evaluation*. 2nd edition. New York / London: Routledge.
- Gobbo, F., & Benini, M. (2011). *Constructive Adpositional Grammars. Foundations of Constructive Linguistics*. Newcastle upon Tyne: Cambridge Scholars Publishing.
- Gobbo, F., & Benini, M. (2013). Dependency and valency. From Structural Syntax to Constructive Adpositional Grammars. In K. Gerdes, E. Hajiov & L. Wanner (Eds.), *Computational Dependency Theory* (pp. 113-135). Amsterdam: IOS Press.
- Searle, J. R. (2010). *Making the social world. The structure of human civilization*. Oxford: Oxford University Press: Oxford.
- Tesnière, L. (1959). *Éléments de Syntaxe Structurale*. Paris: Klincksieck (1959).
- Tesnière, L. (2015). *Elements of Structural Syntax*. English translation of Tesnière (1959). Amsterdam: John Benjamins.
- Wagemans, J. H. M. (2016). Constructing a *Periodic Table of Arguments*. In P. Bondy & L. Benacquista (Eds.), *Argumentation, Objectivity, and Bias: Proceedings of the 11th International Conference of the Ontario Society for the Study of Argumentation (OSSA), 18-21 May 2016* (pp. 1-12). Windsor, ON: OSSA.
- Wagemans, J. H. M. (2017). *Periodic Table of Arguments: The atomic building blocks of persuasive discourse*. 41 pp. Published online December 9, 2017. URL = www.periodic-table-of-arguments.org.
- Wagemans, J. H. M. (2018). *Argument Identification Procedure (AIP) – Version 3*. Published online September 10, 2018. URL = www.periodic-table-of-arguments.org/argument-identification-procedure.