Modeling the Binding of Implicit Arguments in Complement Clauses in ACT-R/Double-R

Jerry T. Ball (Jerry.Ball@wpafb.af.mil)
Air Force Research Laboratory, 711th Human Performance Wing
2620 Q Street, Bldg 852, WPAFB, OH 45433

Abstract

This paper motivates the introduction of grammatical function specific buffers (subject, object), the representation of grammatical features (number, animacy, gender), and the encoding of verb preferences (transitive vs. intransitive; subject vs. object control) in order to model the binding of implicit arguments in complement clauses in Double-R.

Keywords: ACT-R; Double-R; binding; co-reference; implicit argument; complement clause

Introduction

Double-R, which stands for referential and relational, is a computational cognitive linguistic theory of the processing of linguistic expressions into representations which encode grammatically expressed and implied aspects of referential and relational meaning. In addition to being a cognitive linguistic theory (cf. Langacker 1987, 1991), Double-R is implemented as a computational cognitive model (Ball, 2011a) in the ACT-R cognitive architecture (Anderson, 2007). Double-R uses the representation and processing mechanisms of ACT-R and relies on many of the cognitive constraints of the architecture (Ball, 2012, 2013). Double-R also adheres to two well-established cognitive constraints on Human Language Processing (HLP): incremental and interactive processing (Gibson & Pearlmutter, 1998; Altmann & Mirkovic, 2009; Altmann & Steedman, 1988; Tanenhaus et al., 1995). Double-R processes written linguistic expressions incrementally, one word or multi-word unit at a time and interactively and probabilistically uses all available information to make the best choice at each choice point. In addition, Double-R incorporates a non-monotonic mechanism of *context accommodation* which is capable of making modest adjustments to the evolving representation in the right context. The result is a *pseudo-deterministic* language processing capability (Ball, 2011a), a capability that presents the appearance and efficiency of deterministic processing despite the rampant ambiguity that makes truly deterministic processing impossible. Double-R has been under research and development in ACT-R since 2002. It has broad coverage of English language constructions and is approaching the capacity of leading computational linguistic systems, without being tuned to any specific corpus or being limited to syntactic analysis. Currently, the computational implementation consists of more than 1250 productions which express grammatical knowledge and nearly 58,000 declarative memory chunks, most of which represent lexical items.

The current focus of research is on modeling the binding and co-reference of referring expressions within (intra-sentential) and across (inter-sentential) sentences (Ball, in preparation). This paper addresses the modeling of the intra-sentential binding of implicit arguments of complement clauses, motivating the introduction of grammatical function specific buffers (subject, object, indirect object, wh-focus), the representation of grammatical features (definiteness, number, animacy, gender, person), and the encoding of verb preferences (intransitive, transitive, ditransitive, clausal complement; subject control, object control).

Intra-Sentential Binding

Sentences in English often encode multiple, related situations, where each situation is encoded in a separate clause. For example, in “Bill thinks John likes Mary”, the clause “John likes Mary”—which encodes a “liking” situation, is embedded within the matrix clause “Bill thinks John likes Mary”—which encodes a “thinking” situation. The embedding of the complement clause “John likes Mary” within the matrix clause “Bill thinks John likes Mary” is possible because the verb “think” accepts a clausal complement. In this example, “John likes Mary” is a fully specified clause with a tensed verb “likes” and subject and object arguments. However, it is more common for complement clauses to have reduced forms in which the embedded verb lacks tense and the embedded subject and/or object is not expressed. When this happens, the unexpressed elements of the complement clause often depend on the matrix clause to provide their expression. Consider the sentence

I want *to eat*

The clause “to eat” which functions as the clausal complement of “want” does not express either its subject or object argument. Instead, these arguments may be provided by the corresponding subject and object (if there is one) of the matrix clause and there are grammatical mechanisms for determining which matrix clause argument functions as the subject of “to eat” and which may function as the object. For these mechanisms to work, it is necessary to have access to the arguments of the matrix clause when the complement clause is processed. It is also necessary to provide a mechanism for representing the relationship between the unexpressed arguments of the complement clause and the corresponding matrix clause arguments. Double-R provides a *binding* mechanism to express this relationship.

The prediction for a subject in a clause comes from general grammatical principles—every clause has an explicit or implicit subject. In the case of a tensed declarative clause (e.g. "*I* want it"), the subject is explicit. In the case of an imperative clause (e.g. "Give me it!"), the subject is implicitly "you"—the person being addressed. In the case of a subordinate infinitive clause "I want *to eat*"), the subject is implicit and must be recovered from the linguistic context. In the case of a narrative lacking a subject (e.g. "Got up. Brushed my/his/her/their teeth. Took a shower.") the implicit subject must be recovered from the non-linguistic or discourse context.

Whereas the prediction for a subject is based on general grammatical principles, the prediction for an object or indirect object depends on the relational head of a clause: a ditransitive verb sets up the expectation for a direct and indirect object (or recipient), and a transitive verb or preposition sets up an expectation for an object. On the other hand, an adjective or intransitive verb does not set up an expectation for any non-subject arguments.

The prediction for a clausal complement following a relational head and any non-subject arguments (i.e. object, indirect object) comes from a mixture of lexical and general grammatical principles. Many verbs functioning as clausal head require (or at least prefer) a clausal complement and the form of the clausal complement varies from verb to verb. For example, "want" accepts an infinitival complement, but not a participial complement (e.g. "I want to eat" vs. "*I want eating"), whereas "like" accepts both (e.g. "I like to eat"; "I like eating"). In addition, a clausal complement can be added to the vast majority of clauses (regardless of the clausal head) and often corresponds to the expression of the reason for the situation expressed by the clause. In "I want a cookie to eat", "to eat" is an infinitival (clausal) complement that explains why the situation "I want a cookie" exists. Note that "I want a cookie" is grammatical without "to eat" which indicates that the infinitival complement is optional with "want".

ACT-R comes with a collection of buffers (e.g. retrieval buffer, imaginal buffer, goal buffer) that constitute (at least part of) its working memory (Ball, 2012). These buffers have proved inadequate to support language analysis, especially with respect to modeling binding and co-reference, and we have added a collection of language (and grammatical function) specific buffers which include **subject**, **object, indirect object**, **wh-focus**, **relative-focus** and **locative-focus**. The existence of these buffers is motivated on functional grounds. They are needed to support binding and co-reference. Although Taatgen & Anderson (2008) argue on theoretical grounds for limiting functionality and keeping ACT-R tightly constrained, functional considerations are important in the creation of complex cognitive models and may have theoretical implications as well (Ball, 2011b, 2012). Whereas a model which focused on a particular aspect of binding or co-reference might make do with the existing buffers, a broad coverage model like Double-R that is intended to be functional as well as cognitively plausible simply does not have the needed architectural resources. Fortunately, ACT-R 6 supports the addition of buffers (and modules) as a mechanism for extending ACT-R and we have taken advantage of this capability in our research.

The language specific buffers that have been added to ACT-R give the language analysis capabilities of Double-R the flavor of a language module. However, we do not claim that these buffers are fully encapsulated within a language module, and the language analysis productions in the procedural module which access these buffers are interleaved with productions which perform other cognitive functions, and which may also access these buffers. We also do not claim that these buffers are innate. For example, in languages like Chinese (unlike English), wh-words occur in normal argument position and a wh-focus buffer may not be needed. We do claim that humans are capable of learning how to buffer information that may be subsequently needed—a form of expertise. Binding and co-reference in language analysis provide concrete examples of this need.

To see how these language specific buffers are needed to support Double-R's binding mechanism, consider the processing of

1. I want it
2. I want to eat
3. I want you to eat
4. I want the cookie to eat
5. What do you want me to eat

"Want" is a transitive verb that can optionally take an infinitival complement in addition to or in place of the object. When the infinitival complement occurs, the subject of the infinitival complement is not expressed and must be inferred from the matrix clause. There are two possibilities: 1) the subject of the infinitival complement corresponds to the object of the matrix clause (if there is one), or 2) the subject of the infinitival complement corresponds to the subject of the matrix clause. To handle these alternatives, both the subject and object of the matrix clause must be available to support binding by the subject of "to eat". In addition, if the infinitival complement is headed by a transitive verb (e.g. "eat"), the object of the transitive verb may also be unexpressed. In this case, the object may also be inferred from the matrix clause.

First, consider the processing of "I want it". When "I" is processed, a nominal corresponding to "I" is retrieved from memory (or projected from "I"), its referent is determined and the nominal is placed in the subject buffer. (Note that the processing of "I" does not lead to projection of a clause. Language is often used to point out objects in the environment and projecting a clause on the basis of a nominal is not well motivated.) When "want" is processed, a declarative clause is projected based on "want" being a tensed verb and a subject being available in the subject buffer. The nominal in the subject buffer is integrated as the subject of the clause. When "it" is processed, a nominal is retrieved (or projected). The nominal is integrated as the object of the predicate-transitive-verb construction projected from "want" and it is also placed in the object buffer. The resulting representation is shown in Figure 1 (Note that in Double-R the key term is *referring expression*, with two primary subtypes: *object referring expression* which corresponds to nominal or noun phrase, and *situation referring expression* which corresponds to clause. Henceforth, we use the Double-R terms. See the legend on page 5 for a fuller list of terms.)

Figure 1: "I want it" (see page 6)

This example demonstrates the need for a subject buffer to support integration of the subject into the situation referring expression projected by the main verb under the assumption that the subject does not project a situation referring expression by itself (for the reasons discussed above).

The reason the object referring expression is placed in a special subject buffer and not in a subject slot of a chunk in a core ACT-R buffer (e.g. goal, imaginal), is because the grammatical features of the object referring expression need to be accessible to support binding and co-reference. If the object referring expression chunk were placed in a slot of a chunk in a buffer, its grammatical feature slots would not be accessible. The reason the object referring expression is not placed in a core ACT-R buffer where the grammatical feature slots would be accessible is because there are an insufficient number of core buffers to hold all the referring expressions in the matrix clause. The alternative of storing the object referring expression in declarative memory and retrieving it when needed has proved functionally unmanageable for handling multiple and chained long-distance dependencies. For example, in "What do you want the boy on the chair by the table next to the girl *to eat*", binding the subject of "to eat" to "the boy" and the object of "to eat" to "what" leads to severe interference without an object and wh-focus buffer to facilitate this binding.

The processing of "I want you to eat" proceeds similarly to "I want it", up to the processing of "to eat". At this point, the object referring expression retrieved from "I" is in the subject buffer and the object referring expression retrieved from "you" is in the object buffer. The expression "to eat" is processed as a multi-word unit which projects an infinitive clause. The subject of an infinitive clause must be recovered from the linguistic context. In this example, there are two object referring expressions available: the subject and the object of the matrix clause. The grammatical default is to prefer to bind the subject of the infinitive clause to the object of the matrix clause. This default applies so long as the grammatical features of the object are compatible with the subject of the infinitive clause. In particular, the subject of "to eat" is presumed to be animate or human. The pronoun "you" projects the animacy feature "human", so the default applies and the subject of the infinitive clause is bound to "you". To support binding, the subject of "to eat" is represented by an implied object referring expression with head **PRO** (the term **PRO** is borrowed from generative grammar and indicates an implicit subject). The bind index of PRO is set to match the bind index of "you". Although PRO represents the binding from the subject of the infinitive to the matrix subject, the matrix subject itself (not PRO) is placed in an embedded subject buffer which is distinct from the subject buffer, to support further processing. This example demonstrates the need to retain the object of the matrix clause for binding. The resulting representation is shown in Figure 2.

Figure 2: "I want you to eat" (see page 6)

In the processing of "I want the cookie to eat", the animacy feature of the object is not compatible with the subject of the infinitive clause. In this case, the alternative of binding to the subject of the matrix clause is considered. (Actually, these alternatives are considered in parallel based on ACT-R's production matching capability combined with production utility, with the highest utility production which matches the input and context determining the outcome.) Since the animacy of "I" is compatible, the implicit subject of the infinitive clause is bound to the matrix subject. In addition, the object of the matrix clause is available to be bound by the implicit object of the predicate-transitive-verb construction projected by "to eat" and the grammatical features are compatible with that binding. The implicit object of "to eat" is represented as an object referring expression with head **trace** (the term **trace** is also borrowed from generative grammar and indicates a displaced object). This trace element is set to match the bind index of the matrix object and the matrix object is placed in an embedded object buffer which is distinct from the object buffer. This example demonstrates the need to retain both the subject and object of the matrix clause to support binding. The resulting representation is shown in Figure 3.

Figure 3: "I want the cookie to eat" (see page 6)

The grammatical features that get projected from lexical items to referring expressions (Ball, 2010) are crucial for determining binding, as are the argument preferences of the verbs "want" and "eat" which are transitive—indicating the expectation for an object. However, grammatical features are not always definitive. Consider the expression "I want it to eat". The pronoun "it" can be used to refer to either animate (e.g. dog) or inanimate (e.g. cookie) objects (and even humans when their sex is unknown as is often the case with babies). In the case of "it", binding and co-reference depend on the actual referent. If the referent of "it" is a cookie, then binding the object to "it" is preferred; if the referent is an animal or human, then binding the subject to "it" is preferred. In the absence of an identified referent, the binding is ambiguous. By default, Double-R treats "it" as animate and binds the subject. Double-R doesn't currently have the capability to use the referent of a referring expression to determine binding in ambiguous cases.

To motivate the need for retaining the indirect object in a buffer, consider the processing of the expression "I gave him the cookie to eat". In this example, the processing of "him" leads to retrieval of an object referring expression which is integrated as the indirect object of the predicate-ditransitive-verb construction projected by "gave". This object referring expression is also placed in the indirect object buffer. At the processing of "to eat", the default preference is to bind the implied subject of the infinitive clause to the indirect object which is normally animate or human. It is also preferred to bind the (direct) object to the implied object of the transitive verb “eat”. The resulting representation is shown in Figure 4.

Figure 4: “I gave him the cookie to eat” (see page 6)

The processing of intra-sentential infinitival complements provides strong motivation for retaining object referring expressions which function as arguments in the matrix clause in buffers to support binding by implied arguments in the subordinate clause. An earlier version of the model made use of a fixed size stack of object referring expressions, but lacked grammatical function specific buffers. This architecture proved to be functionally inadequate. In the previous example, it is possible for the object referring expressions for “I”, “him” and “the cookie” to be stacked such that the object “the cookie” is on top, “the indirect object “him” is next and the subject “I” is on the bottom. While a stack will handle this example, it does not generalize to more complex examples. Consider “I gave him the book on the table in the kitchen to read”. If all object referring expressions (e.g. “I”, “him”, “the book”, “the table”, “the kitchen”) are stacked, then it is not possible to determine the grammatical function of the object referring expressions based on position in the stack. Further, if the stack is fixed in size (an unbounded stack is cognitively implausible), it is always possible to generate a linguistic expression which will cause the stack to overflow leading to the loss of a referring expression that is needed for subsequent binding. Of course, this would be OK if it matched empirical findings, but it doesn’t appear to. On the other hand, the stack of object referring expressions is still needed to support the integration of post-head modifiers. In the example, “in the kitchen” modifies “the table”, and “on the table” modifies “the book”. A fixed size stack on the order of 3 or 4 object referring expressions seems a cognitively reasonable mechanism for handling post-head modifiers which typically modify the preceding object referring expression, but may also modify earlier expressions (e.g. in “I saw the man on the hill with the binoculars”, “with the binoculars” may modify “saw”, “the man” or “the hill”, although modifying “the hill” is semantically dispreferred). The current model combines grammatical function specific buffers with a stack of the most recent object referring expressions. Besides being functionally motivated, this architecture is compatible with empirical evidence of primacy and recency effects. The grammatical function specific buffers retain the outermost object referring expression in a deeply modified expression—supporting primacy effects, while the fixed stack retains the 3 most recent object referring expressions—supporting recency effects. It is important to note that in this architecture, an object referring expression may constitute the contents of more than one buffer. In the example in Figure 4, “the cookie” fills the object buffer as well as the most recent object referring expression buffer in the stack. In a sense, the buffers provide pointers to object referring expressions, except that the contents of the object referring expression are directly accessible in the buffer without a retrieval from declarative memory.

Wh Questions

The processing of wh-questions demonstrates the need for a wh-focus buffer to support binding. Consider the expression “What do you want me to eat?”. The processing of “what” leads to projection of a wh object referring expression that is put in the wh-focus buffer. Note that the processing of “what” does not lead to projection of a wh-question. (There are wh-constructions like “what he said…is true” that are not wh-questions.) The processing of the auxiliary verb “do” in the context of a wh object referring expression in the wh-focus buffer leads to projection of a wh-question with a wh-focus function that is filled by the referring expression in the wh-focus buffer and an operator function that is filled by “do”. The processing of “you” following “do” results in retrieval of an object referring expression that is integrated as the subject of the wh-question and this object referring expression is also placed in the subject buffer. The processing of “want” leads to projection of a predicate transitive verb construction that is integrated as the head of the wh-question. In addition, an implied trace object of “want” is created and bound to the wh object referring expression in the wh-focus buffer. The binding of the trace object to the wh-focus reflects Double-R’s greedy mechanism for modeling long distance dependencies involving fronted wh words. Note that if the entire input were “What do you want?”, the binding of the implied object of “want” to the wh-focus is expected.

Figure 5: Result after “What do you want…” (see page 6)

The processing of “me” leads to retrieval of an object referring expression. This referring expression is integrated as the object of “want” displacing the implied trace that was bound to the wh-focus. This displacement is an example of the context accommodation mechanism at work. The processing of “to eat” leads to projection of an infinitive situation referring expression. An implicit PRO object referring expression is projected and bound to the matrix object “me”. In addition a trace object referring expression is projected and integrated as the object of “to eat”. This trace expression is bound to the wh-focus.

Figure 6: Result after “What do you want me to eat?” (p. 6)

Control

The examples above focus on the importance of representing grammatical features and verb argument preferences for determining the binding of implicit arguments in complement clauses associated with the main verb “want”. There is an additional contrast between the behavior of verbs like “want” (*object control* verbs, or better, object-to-subject control) and verbs like “promise” (*subject control* verbs, or better subject-to-subject control) which affects the binding of implicit arguments. Control is a central topic in modern linguistic theory (cf. Culicover, 2009). Consider the following classic examples from Chomsky (1981):

He persuaded me to go
He promised me to go

"Persuade" is an object control verb: the object of "persuade" determines the binding of the implicit subject of the infinitival clause "to go". This is the default behavior discussed above for "want". "Promise" is a subject control verb: the subject of "promise" determines the binding of the implicit subject. Subject control is the exception for verbs—only a few verbs exhibit this preference. Control is not limited to verbs. Adjectives functioning as predicates also exhibit control. Consider

He is eager to please
He is easy to please

(also from Chomsky, 1981). Subject-to-subject control is the default for (predicate) adjectives. The subject of "eager" determines the implicit subject of "to please". "Easy" is exceptionally a subject-to-object control adjective: the subject of "easy" determines the implicit object of "to please" and the implicit subject of "to please" is unbound (e.g. "He is easy *for someone* to please"). The examples with adjectives also demonstrate the possibility of adding an optional clausal complement to clauses containing a predicate adjective, despite the fact that adjectives do not normally expect a complement. "He is eager" and "he is easy" are both grammatical without the infinitival complement.

Conclusions

This paper motivates the introduction of grammatical function specific buffers (subject, object), the representation of grammatical features (number, animacy, gender), and the encoding of verb preferences (transitive vs. intransitive; subject control vs. object control) in order to model the binding of implicit arguments of complement clauses within Double-R, a cognitive linguistic theory of human language processing implemented as a cognitive model in ACT-R.

References

Altmann, G. & Mirkovic, J. (2009). Incrementality and Prediction in Human Sentence Processing. *Cognitive Science*, 222, 583-609.

Altmann, G., & Steedman, M. (1988). Interaction with context during human sentence processing. *Cognition*, *30*, 191-238.

Anderson, J. (2007). *How Can the Human Mind Occur in the Physical Universe?* NY: Oxford University Press.

Ball, J. (in preparation). The Representation and Processing of Referring Expressions in Double-R—Binding and Co-reference.

Ball, J. (2013). Advantages of ACT-R over Prolog for Natural Language Analysis. *Proceedings of the 22nd Annual Conference on Behavior Representation in Modeling and Simulation*.

Ball, J. (2012). Explorations in ACT-R Based Language Analysis – Memory Chunk Activation, Retrieval and Verification without Inhibition. In N. Russwinkel, U. Drewitz & H. van Rijn (eds), *Proceedings of the 11th International Conference on Cognitive Modeling*, 131-136. Berlin: Universitaets der TU Berlin.

Ball, J. (2011a). A Pseudo-Deterministic Model of Human Language Processing. In L. Carlson, C. Hölscher, & T. Shipley (Eds.), *Proceedings of the 33rd Annual Conference of the Cognitive Science Society*, 495-500. Austin, TX..

Ball, J. (2011b). Explorations in ACT-R Based Cognitive Modeling – Chunks, Inheritance, Production Matching and Memory in Language Analysis. *Proceedings of the AAAI Fall Symposium: Advances in Cognitive Systems*, 10-17. AAAI.

Ball, J. (2010). Projecting Grammatical Features in Nominals: Cognitive Processing Theory & Computational Implementation. *Proceedings of the 19th Annual Conference on Behavior Representation in Modeling and Simulation*. Charleston, SC.

Ball, J. (2007). A Bi-Polar Theory of Nominal and Clause Structure and Function. *Annual Review of Cognitive Linguistics*, 27-54. Amsterdam: John Benjamins.

Chomsky, N. (1981). *Lectures on Government and Binding*. Dordrecht: Foris Publications.

Culicover, P. (2009). *Natural Language Syntax*. Oxford.

Eisenbach, A. & Eisenbach, M. (2006). phpSyntaxTree tool, http://ironcreek.net/phpsyntaxtree.

Gibson, E., & Pearlmutter, N. (1998). Constraints on sentence comprehension. *Trends in Cognitive Sciences*, *2(*7), 262-268.

Heiberg, A., Harris, J., & Ball, J. (2007). Dynamic Visualization of ACT-R Declarative Memory Structure. *Proceedings of the 8th International Conference on Cognitive Modeling*, 233-234. Oxford, UK: Taylor & Francis/Psychology Press.

Langacker, R. (1987,1991). *Foundations of Cognitive Grammar, Vols 1 and 2*. Stanford.

Taatgen, N. & Anderson, J. (2008). ACT-R. In R. Sun (ed.), *Constraints in Cognitive Architectures*. Cambridge University Press, 170-185.

Tanenhaus, M., Spivey-Knowlton, M., Eberhard, K., & Sedivy, J. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science, 268*,1632-1634.

Diagrams

The diagrams below were automatically generated from the output of the language analysis model using code developed in house that generates bracket structures from ACT-R chunks (Heiberg, Harris & Ball, 2007) and submits the bracket structures to the phpSyntaxTree software (Eisenbach & Eisenbach, 2006). Please zoom in to better view the diagrams.

Legend:

object referring expression (obj-refer-expr) ~ nominal or NP
  pronoun object referring expression (pron-obj-refer-expr)
situation referring expression (sit-refer-expr) ~ clause
  declarative situation referring expression (decl-sit-refer-expr)
  infinitive situation referring expression (inf-sit-refer-expr)
predicate transitive verb (pred-trans-verb)
personal pronoun (pers-pron)
subject (subj), object (obj), indirect object (iobj), complement
discourse function (df) – statement (stmt), question (quest)
definiteness – definite (def), indefinite (indef)
number – singular (sing), plural (plur); gender – male, female
person – first, second, third; case – subject (subj), object (obj)
animacy (animate) – human, animate, inanimate
tense (level 1 – tense-1) – finite (fin), non-finite (non-fin)
tense (level 2 – tense) – present (pres), past
voice – active (act), passive (pass), inactive (inact)

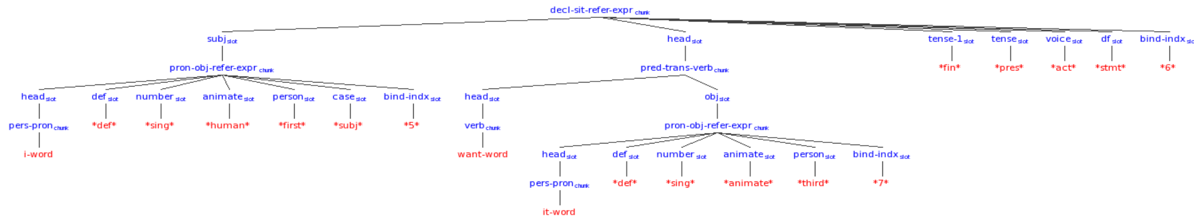

Figure 1: Result after processing “I want it”

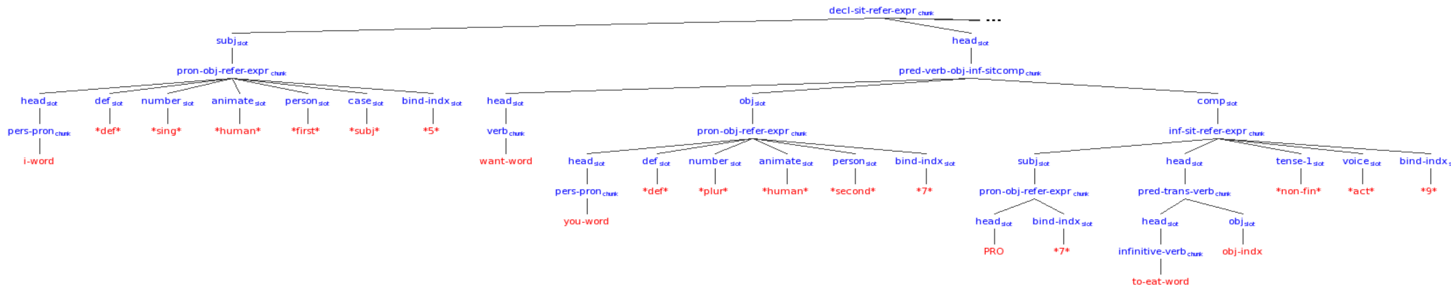

Figure 2: Result after processing “I want you to eat”

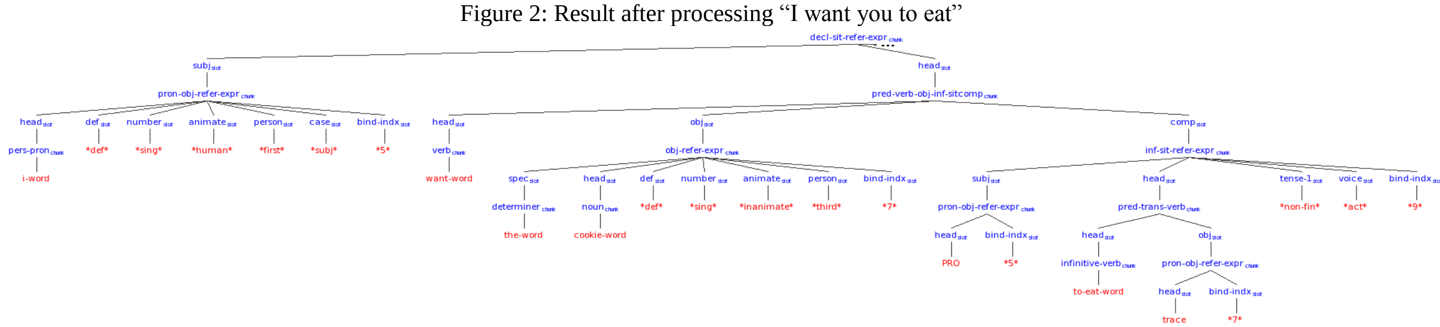

Figure 3: Result after processing “I want the cookie to eat”

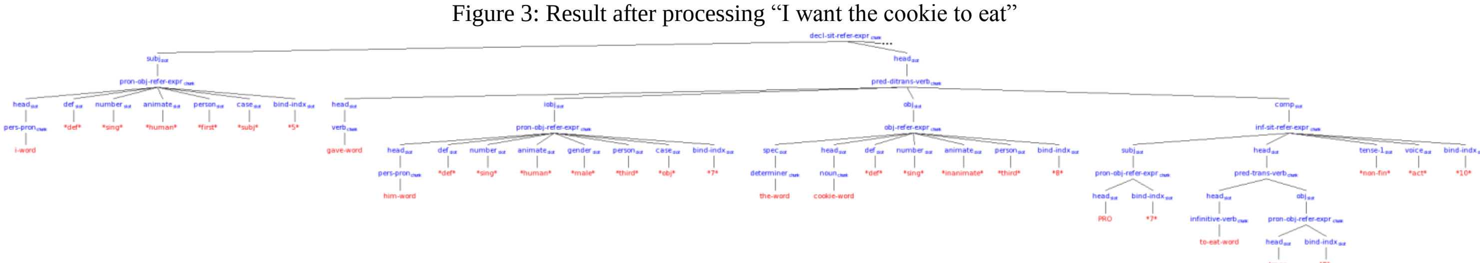
Figure 4: Result after processing “I gave him the cookie to eat”

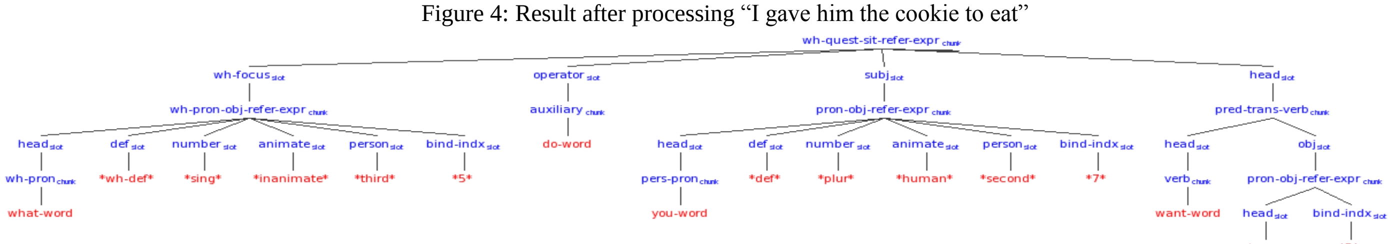

Figure 5: Result after processing “What do you want…”

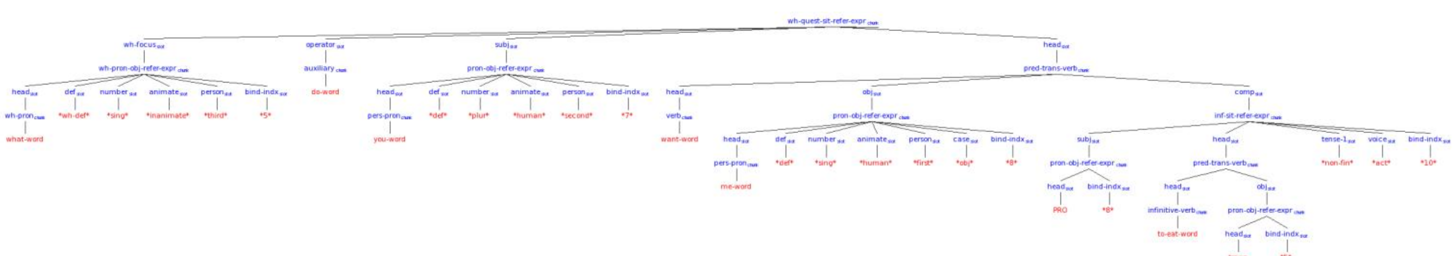
Figure 6: Result after processing “What do you want me to eat”