The Metagrammar Goes Multilingual: A Cross-Linguistic Look at the V2-Phenomenon

Alexandra Kinyon
Department of CIS
University of Pennsylvania
kinyon@linc.cis.upenn.edu

Owen Rambow
CCLS
Columbia University
rambow@cs.columbia.edu

Tatjana Scheffler
Department of Linguistics
University of Pennsylvania
tatjana@ling.upenn.edu

SinWon Yoon
UFRL
Université Paris 7
swyoon@linguist.jussieu.fr

Aravind K. Joshi
Department of CIS
University of Pennsylvania
joshi@linc.cis.upenn.edu

Abstract

We present an initial investigation into the use of a metagrammar for explicitly sharing abstract grammatical specifications among languages. We define a single class hierarchy for a metagrammar which allows us to automatically generate grammars for different languages from a single compact metagrammar hierarchy. We use as our linguistic example the verb-second phenomenon, which shows considerable variation while retaining a basic property, namely the fact that the verb can appear in one of two positions in the clause.

1 An Overview of Metagrammars

A metagrammar (MG) factors common properties of TAG elementary trees to avoid redundancy, ease grammar development, and expand coverage with minimal effort: typically, from a compact manually encoded MG of a few dozen classes, one or more TAGs with several hundreds of elementary trees are automatically generated. This is appealing from a grammar engineering point of view, and also from a linguistic point of view: cross-linguistic generalizations are expressed directly in the MG. In this paper, we extend some earlier work on multilingual MGs (Candito, 1998; Kinyon and Rambow, 2003) by proposing cross-linguistic and framework-neutral syntactic invariants, which we apply to TAG. We focus on the verb-second phenomenon as a prototypical example of cross-language variation.

The notion of Metagrammar Metagrammars were first introduced by Candito (1996) to manually encode syntactic knowledge in a compact and abstract class hierarchy which supports multiple inheritance, and from which a TAG is automatically generated offline. Candito's class hierarchy imposes a general organization of syntax into three dimensions:

- Dimension 1: to encode initial subcategorization frames i.e. TAG tree families
- Dimension 2: to encode valency alternations / redistribution of syntactic functions
- Dimension 3: to encode the surface realization of arguments.

Each class in the MG hierarchy is associated with a partial tree description The tool computes a set of well-formed classes by combining exactly one terminal class from dimension 1, one terminal class from dimension 2, and n terminal classes from dimensions 3 (n being the number of arguments subcategorized by the lexical head anchoring the elementary tree(s) generated). The conjunction of the tree descriptions associated with each well-formed class in the set yields a minimal satisfying description, which results in the generation of one or more elementary trees. Candito's tool was used to develop a large TAG for French as well as a medium-size TAG for Italian Candito (1999), so multilinguality was addressed from the start, but each language had its dedicated hierarchy, with no sharing of classes despite the obvious similarities between Italian and French. A related approach was proposed by (Xia, 2001); the work of Evans, Gazdar, and Weir (2000) also has some common elements with MG.

Framework- and language-neutral syntactic invariants Using a MG, and following Candito, we can postulate cross-linguistic and cross-framework syntactic invariants such as:

Proceedings of the 8th International Workshop on Tree Adjoining Grammar and Related Formalisms, pages 17–24,
Sydney, July 2006.

- The notion of subcategorization
- The existence of a finite number of syntactic functions (subject, object etc.)
- The existence of a finite number of syntactic categories (NP, PP, etc.)
- The existence of valency alternations (Candito's dimension 2)
- The existence, orthogonal to valency alternations, of syntactic phenomena which do not alter valency, such as *wh*-movement (Candito's dimension 3).

These invariants — unlike other framework-specific syntactic assumptions such as the existence of "movement" or "*wh*-traces" — are accepted by most if not all existing frameworks, even though the machinery of a given framework may not necessarily account explicitly for each invariant. For instance, TAG does not have an explicit notion of syntactic function: although by convention node indices tend to reflect a function, it is not enforced by the framework's machinery.[1]

Hypertags Based on such framework- and language-neutral syntactic properties, Kinyon (2000) defined the notion of **Hypertag** (HT), a combination of Supertags (ST) Srinivas (1997) and of the MG. A ST is a TAG elementary tree, which provides richer information than standard POS tagging, but in a framework-specific manner (TAG), and also in a grammar-specific manner since a ST tagset can't be ported from one TAG to another TAG. A HT is an abstraction of STs, where the main syntactic properties of any given ST is encoded in a general readable Feature Structure (FS), by recording which MG classes a ST inherited from when it was generated. Figure 1 illustrates the <ST, HT> pair for *Par qui sera accompagnée Marie* 'By whom will Mary be accompanied'. We see that a HT feature structure directly reflects the MG organization, by having 3 features "Dimension 1", "Dimension 2" and "Dimension 3", where each feature takes its value from the MG terminal classes used to generate a given ST.

The XMG Tool Candito's tool brought a significant linguistic insight, therefore we essentially retain the above-mentioned syntactic invariants. However, more recent MG implementations have been developed since, each adding its significant contribution to the underlying metagrammatical hypothesis.

In this paper, we use the eXtensible MetaGrammar (XMG) tool which was developed by Crabbé

[1]But several attempts have been made to explicitly add functions to TAG, e.g. by Kameyama (1986) to retain the benefits of both TAG and LFG, or by Prolo (2006) to account for the coordination of constituents of different categories, yet sharing the same function.

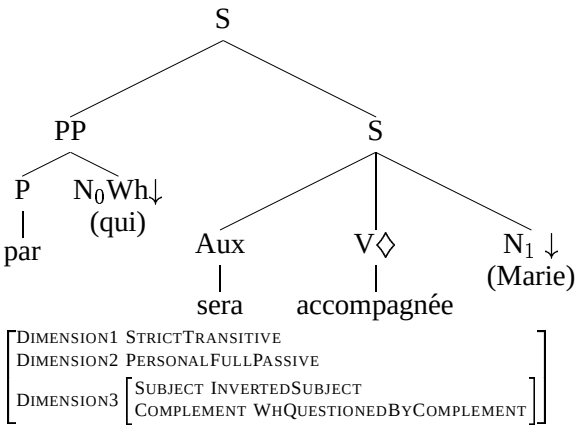

Figure 1: A <SuperTag, HyperTag> pair for *accompagnée* ('accompanied') obtained with Candito's MetaGrammar compiler

(2005). In XMG, an MG consists of a set of *classes* similar to those in object-oriented programming, which are structured into a multiple inheritance hierarchy. Each class specifies a partial tree description (expressed by dominance and precedence constraints). The nodes of these tree fragment descriptions may be annotated with features. Classes may instantiate each other, and they may be parametrized (e.g., to hand down features like the grammatical function of a substitution node). The compiler unifies the instantiations of tree descriptions that are called. This unification is additionally guided by *node colors*, constraints that specify that a node must not be unified with any other node (red), must be unified (white), or may be unified, but only with a white node (black). XMG allows us to implement a hierarchy similar to that of Candito, but it also allows us to modify and extend it, as no structural assumptions about the class hierarchy are hard-coded.

2 The V2 Phenomenon

The Verb-Second (V2) phenomenon is a well-known set of data that demonstrates small-scale cross-linguistic variation. The examples in (1) show German, a language with a V2-constraint: (1a) is completely grammatical, while (1b) is not. This is considered to be due to the fact that the finite verb is required to be located in "second position" (V2) in German. Other languages with a V2 constraint include Dutch, Yiddish, Frisian, Icelandic, Mainland Scandinavian, and Kashmiri.

(1) a. Auf dem Weg sieht der Junge eine Ente.
on the path sees the boy a duck
'On the path, the boy sees a duck.'

b. * Auf dem Weg der Junge sieht eine Ente.
on the path the boy sees a duck
Int.: 'On the path, the boy sees a duck.'

Interestingly, these languages differ with respect to how exactly the constraint is realized. Rambow and Santorini (1995) present data from the mentioned languages and provide a set of parameters that account for the exhibited variation. In the following, for the sake of brevity, we will confine the discussion to two languages: German, and Yiddish. The German data is as follows (we do not repeat (1a) from above):

(2) a. Der Junge sieht eine Ente auf dem Weg.
the boy sees a duck on the path
'On the path, the boy sees a duck.'
b. ..., dass der Junge auf dem Weg eine Ente
..., that the boy on the path a duck
sieht.
sees
'..., that the boy sees a duck on the path.'
c. Eine Ente sieht der Junge.
a duck sees the boy
'The boy sees a duck.'

The Yiddish data:

(3) a. Dos yingl zet oyfn veg a katshke.
the boy sees on-the path a duck
'On the path, the boy sees a duck.'
b. Oyfn veg zet dos yingl a katshke.
on-the path sees the boy a duck.
'On the path, the boy sees a duck.'
c. ..., az dos yingl zet a katshke
..., that the boy sees a duck
'..., that the boy sees a duck.'

While main clauses exhibit V2 in German, embedded clauses with complementizers are verb-final (2b). In contrast, Yiddish embedded clauses must also be V2 (3c).

3 Handling V2 in the Metagrammar

It is striking that the basic V2 phenomenon is the same in all of these languages: the verb can appear in either its underlying position, or in second position (or, in some cases, third). We claim that what governs the appearance of the verb in these different positions (and thus the cross-linguistic differences) is that the heads—the verbal head and functional heads such as auxiliaries and complementizers—interact in specific ways. For example, in German a complementizer is not compatible with a verbal V2 head, while in Yiddish it is. We express the interaction among heads by assigning the heads different values for a set of features. Which heads can carry which feature values is a language-specific parameter. Our implementation is based on the previous pen-and-pencil analysis of Rambow and Santorini (1995), which we have modified and extended.

The work we present in this paper thus has a threefold interest: (1) we show how to handle an important syntactic phenomenon cross-linguistically in a MG framework; (2) we partially validate, correct, and extend a previously proposed linguistically-motivated analysis; and (3) we provide an initial fragment of a MG implementation from which we generate TAGs for languages which are relatively less-studied and for which no TAG currently exists (Yiddish).

4 Elements of Our Implementation

In this paper, we only address verbal elementary trees. We define a verbal realization to be a combination of three classes (or "dimensions" in Candito's terminology): a *subcategorization frame*, a *redistribution of arguments/valency alternation* (in our case, voice, which we do not further discuss), and a *topology*, which encodes the position and characteristics of the verbal head. Thus, we reinterpret Candito's "Dimension 3" to concentrate on the position of the verbal heads, with the different argument realizations (topicalized, base position) depending on the available heads, rather than defined as first-class citizens. The subcat and argument redistributions result in a set of structures for *arguments* which are left- or right-branching (depending on language and grammatical function). Figure 2 shows some argument structures for German. The topology reflects the basic clause structure, that is, the distribution of arguments and adjuncts, and the position of the verb (initial, V2, final, etc.). Our notion of sentence topology is thus similar to the notion formalized by Gerdes (2002). Specifically, we see positions of arguments and adjuncts as defined by the positions of their verbal heads. However, while Gerdes (2002) assumes as basic underlying notions the fields created by the heads (the traditional *Vorfeld* for the topicalized element and the *Mittelfeld* between the verb in second position and the verb in clause-final position), we only use properties of the heads. The fields are epiphenomenal for us.As mentioned above, we use the following set of features to define our MG topology:

- I (finite tense and subject-verb agreement): creates a specifier position for agreement which must be filled in a derivation, but allows recursion (i.e., adjunction at IP).
- Top (topic): a feature which creates a specifier position for the topic (semantically represented in a lambda abstraction) which must be filled in a derivation, and which does not allow recursion.
- M (mood): a feature with semantic content (to be defined), but no specifier.
- C (complementizer): a lexical feature introduced only by complementizers.

We can now define our topology in more detail. It consists of two main parts:

German:

	What	Features Introduced	Directionality
1	Verb (clause-final)	+I	head-final
2	Verb (V2, subject-inital)	+M, +Top, +I	head-initial
3	Verb (V2, non-subject-initial)	+M, +Top	head-initial
4	Complementizer	+C, +M	head-initial

Yiddish:

	What	Features Introduced	Directionality
1	Verb	+I	head-initial
2	Verb (V2, subject-inital)	+M, +Top, +I	head-initial
3	Verb (V2, non-subject-initial)	+M, +Top	head-initial
4	Complementizer	+C	head-initial

Figure 4: Head inventories for German and Yiddish.

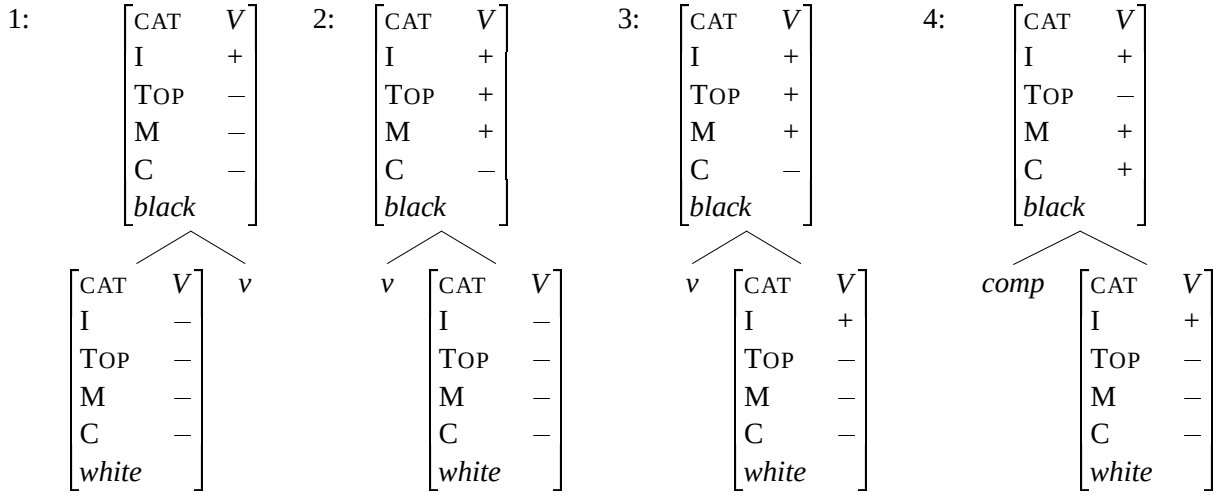

Figure 5: Head structures for German corresponding to the table in Figure 4 (above)

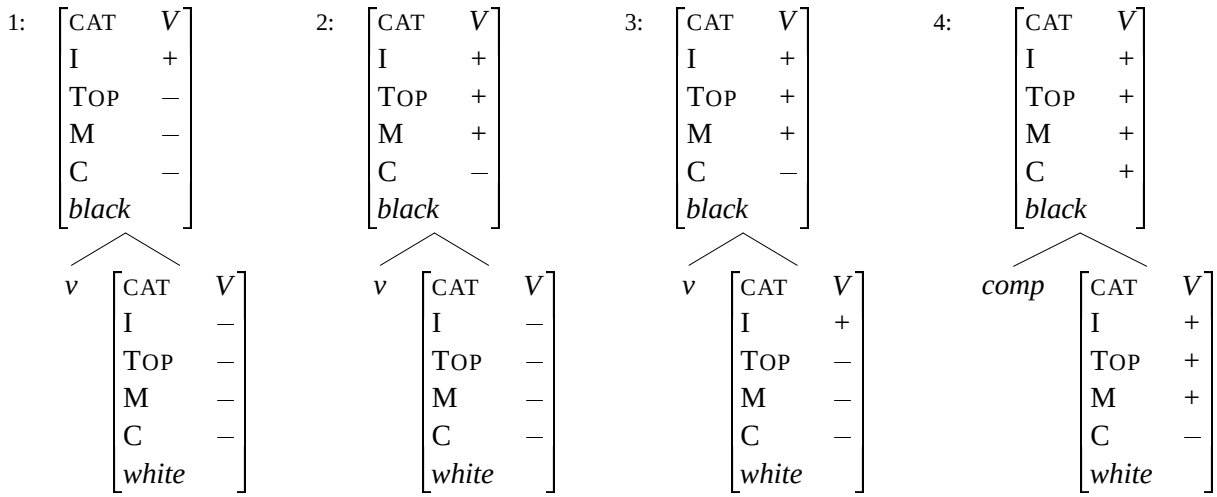

Figure 6: Head structures for Yiddish corresponding to the table in Figure 4 (below)

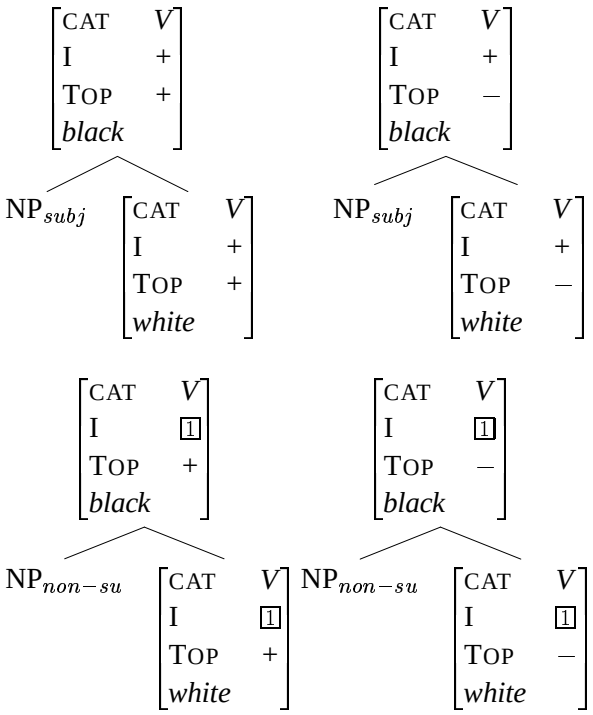

Figure 2: The argument structures

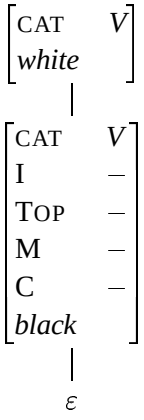

Figure 3: The projection structure; feature values can be filled in at the top feature structure to control the derivation.

- The **projection** includes the origin of the verb in the phrase structure (with an empty head since we assume it is no longer there) and its maximal projection. It is shown in Figure 3. The maximal projection expresses the expected feature content. For example, if we want to model non-finite clauses, the maximal projection will have [−I], while root V2 clauses will have [+Top], and embedded finite clauses with complementizers will have [+I,+C].
- Structures for **heads**, which can be head-initial or head-final. They introduce categorial features. Languages differ in what sort of heads they have. Which heads are available for a given language is captured in a **head inventory**, i.e., a list of possible heads for that language (which use the head structure just mentioned). Two such lists are shown in Figure 4, for German and Yiddish. The corresponding head structures are shown in Figures 5 and 6.

A topology is a combination of the projection and any combination of heads allowed by the language-specific head inventory. This is hard to express in XMG, so instead we list the specific combinations allowed. One might ask how we derive trees for language without the V2 phenomenon. Languages without V2 will usually have a smaller set of possible heads. We are working on a metagrammar for Korean in parallel with our work on the V2 languages. Korean is very much like German without the V2 phenomenon: the verbal head can only be in clause-final position (i.e., head 1 from Figure 5. However, passivization and scrambling can be treated the same way in Korean and German, since these phenomena are independent of V2.

5 Sample Derivation

Given a feature ordering (C > M > Top > I) and language-specific head inventories as in Figure 4, we compile out MGs for German (Figure 5) and Yiddish (Figure 6).[2] The projection and the argument realizations do not differ between the two languages: thus, these parts of the MG can be reused. The features, which were introduced for descriptive reasons, now guide the TAG compilation: only certain heads can be combined. Furthermore, subjects and non-subjects are distinguished, as well as topicalized and non-topicalized NPs (producing 4 kinds of arguments so far). The compiler picks out any number of compatible elements from the Metagrammar and performs the unifications of nodes that are permitted (or required) by

[2] All terminal nodes are "red"; spine nodes have been annotated with their color.

the node descriptions and the colors. By way of example, the derivations of elementary trees which can be used in a TAG analysis of German (2c) and Yiddish (3c) are shown in Figures 7 and 8, respectively.

6 Conclusion and Future work

This paper showed how cross-linguistic generalizations (in this case, V2) can be incorporated into a multilingual MG. This allows not only the reuse of MG parts for new (often, not well-studied) languages, but it also enables us to study small-scale parametric variation between languages in a controlled and formal way. We are currently modifying and extending our implementation in several ways.

The Notion of Projection In our current approach, the verb is never at the basis of the projection, it has always been removed into a new location. This may seem unmotivated in certain cases, such as German verb-final sentences. We are looking into using the XMG unification to actually place the verb at the bottom of the projection in these cases.

Generating Top and Bottom Features The generated TAG grammar currently does not have top and bottom feature sets, as one would expect in a feature-based TAG. These are important for us so we can force adjunction in adjunct-initial V2 sentences (where the element in clause-initial position is not an argument of the verb). We intend to follow the approach laid out in Crabbé (2005) in order to generate top and bottom feature structures on the nodes of the TAG grammar.

Generating test-suites to document our grammars Since XMG offers more complex object-oriented functionalities, including instances, and therefore recursion, it is now straightforward to directly generate parallel multilingual sentences directly from XMG, without any intermediate grammar generation step. The only obstacle remains the explicit encoding of Hypertags into XMG.

Acknowledgments

We thank Yannick Parmentier, Joseph Leroux, Bertrand Gaiffe, Benoit Crabbé, the LORIA XMG team, and Julia Hockenmaier for their invaluable help; Eric de la Clergerie, Carlos Prolo and the Xtag group for their helpful feedback, comments and suggestions on different aspects of this work; and Marie-Hélène Candito for her insights. This work was supported by NSF Grant 0414409 to the University of Pennsylvania.

References

Candito, M. H. 1998. Building parallel LTAG for French and Italian. In *Proc. ACL-98*. Montreal.

Candito, M.H. 1996. A principle-based hierarchical representation of LTAGs. In *Proc. COLING-96*. Copenhagen.

Candito, M.H. 1999. Représentation modulaire et paramétrable de grammaires électroniques lexicalisées. Doctoral Dissertation, Univ. Paris 7.

Clément, L., and A. Kinyon. 2003. Generating parallel multilingual LFG-TAG grammars using a MetaGrammar. In *Proc. ACL-03*. Sapporo.

Clergerie, E. De La. 2005. From metagrammars to factorized TAG/TIG parsers. In *IWPT-05*. Trento.

Crabbé, B. 2005. Représentation informatique de grammaires fortement lexicalisées. Doctoral Dissertation, Univ. Nancy 2.

Evans, R., G. Gazdar, and D. Weir. 2000. Lexical rules are just lexical rules. In *Tree Adjoining Grammars*, ed. A. Abeillé and O. Rambow. CSLI.

Gerdes, K. 2002. DTAG. attempt to generate a useful TAG for German using a metagrammar. In *Proc. TAG+6*. Venice.

Kameyama, M. 1986. Characterising LFG in terms of TAG. In *Unpublished report*. Univ. of Pennsylvania.

Kinyon, A. 2000. Hypertags. In *Proc. COLING-00*. Sarrebrucken.

Kinyon, A., and O. Rambow. 2003. Generating cross-language and cross-framework annotated test-suites using a MetaGrammar. In *Proc. LINC-EACL-03*. Budapest.

Prolo, C. 2006. Handling unlike coordinated phrases in TAG by mixing Syntactic Category and Grammatical Function. In *Proc. TAG+8*. Sidney.

Rambow, Owen, and Beatrice Santorini. 1995. Incremental phrase structure generation and a universal theory of V2. In *Proceedings of NELS 25*, ed. J.N. Beckman, 373–387. Amherst, MA: GSLA.

Srinivas, B. 1997. Complexity of lexical descriptions and its relevance for partial parsing. Doctoral Dissertation, Univ. of Pennsylvania.

Xia, F. 2001. Automatic grammar generation from two perspectives. Doctoral Dissertation, Univ. of Pennsylvania.

XTAG Research Group. 2001. A lexicalized tree adjoining grammar for English. Technical Report IRCS-01-03, IRCS, University of Pennsylvania.

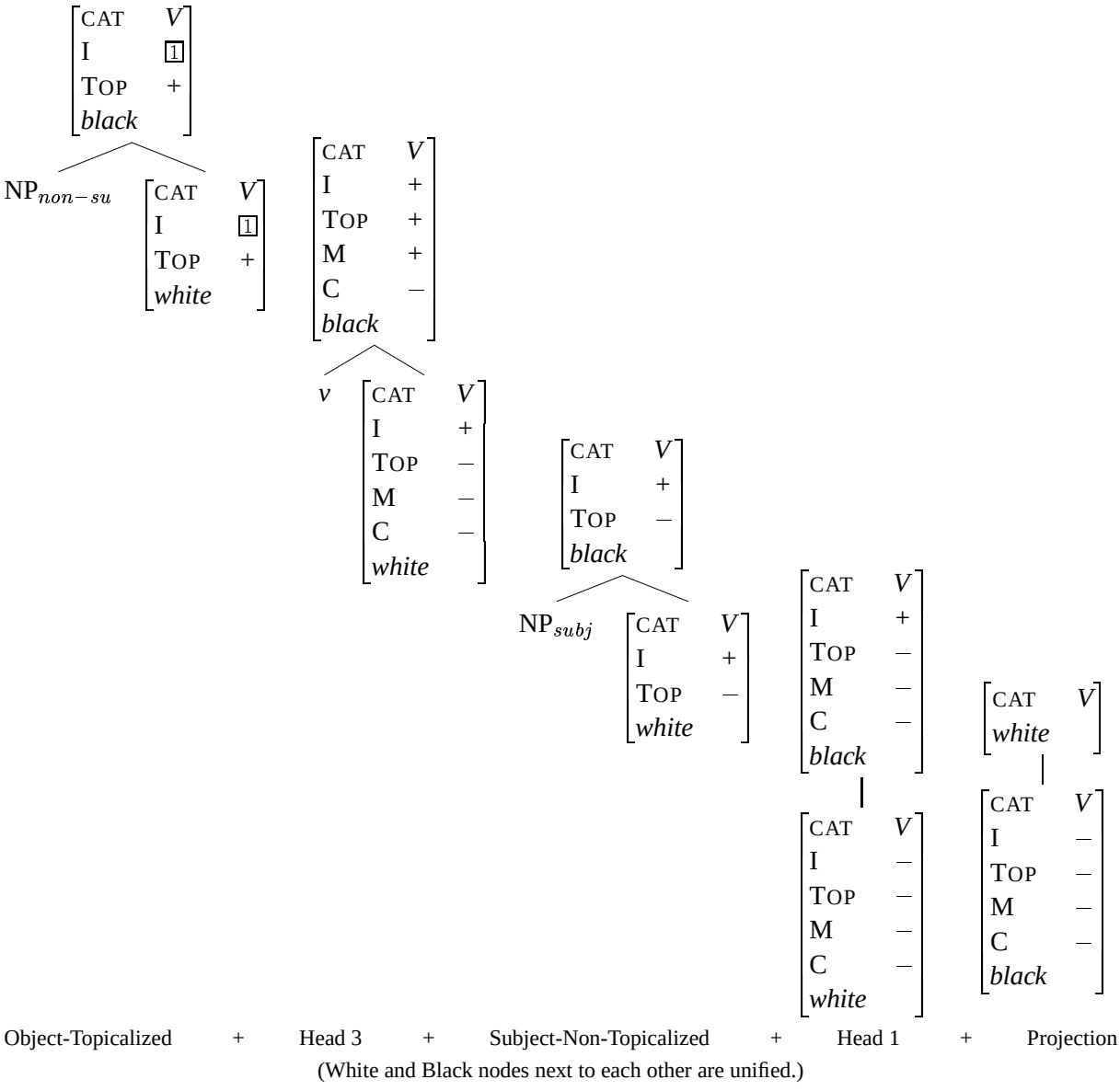

Figure 7: Derivation of the German elementary tree NP_{obj} V NP_{subj} (2d).

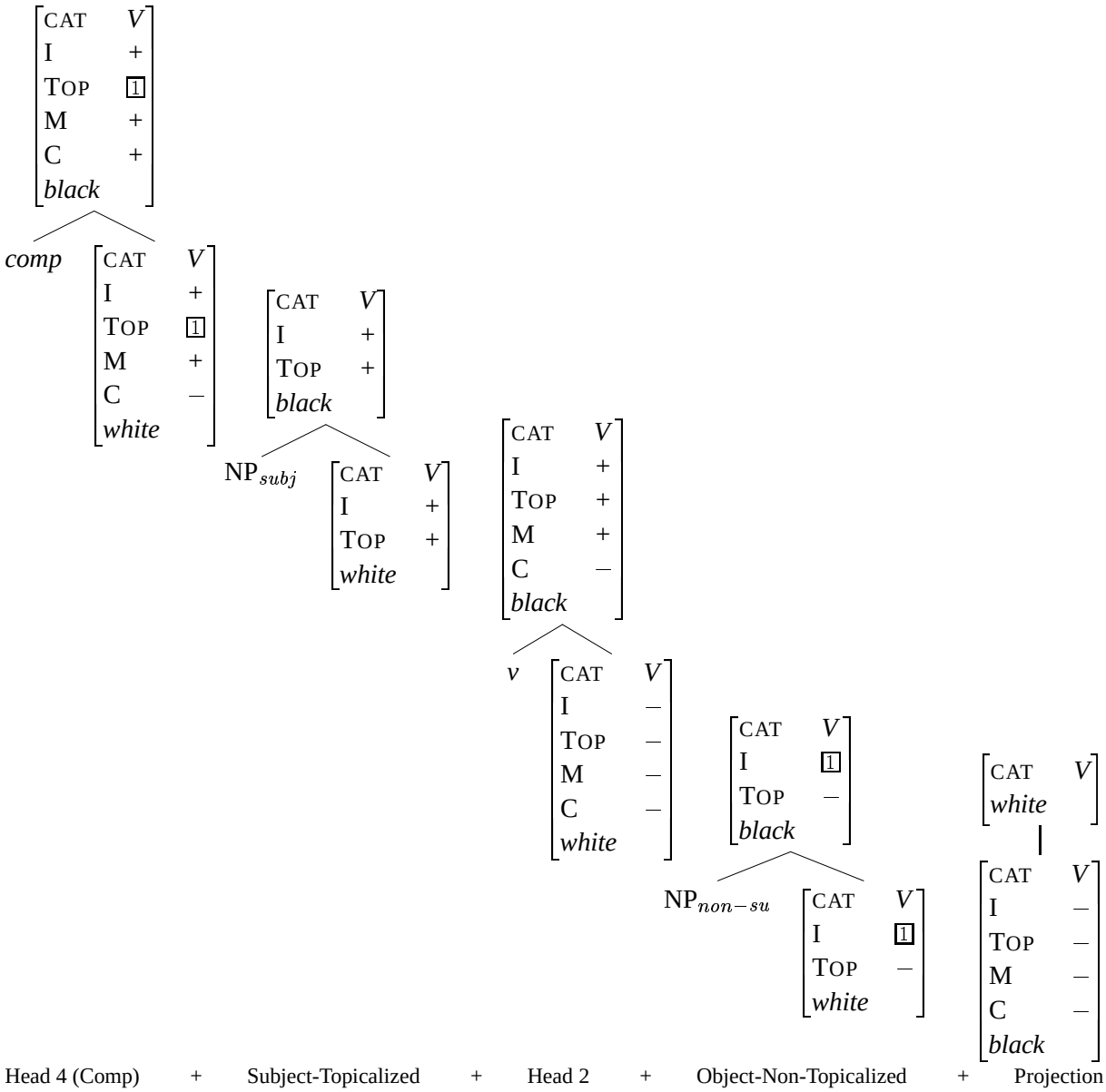

Figure 8: Derivation of the Yiddish elementary tree Comp NP$_{subj}$ V NP$_{obj}$ (3c).