Gontzal Aldai* and Søren Wichmann

Statistical observations on hierarchies of transitivity

https://doi.org/10.1515/flin-2018-0006
Submitted March 24, 2016; Revision invited July 19, 2016;
Revision received February 20, 2017; Accepted February 12, 2018

Abstract: In this paper we first test whether there is statistical support for a transitivity hierarchy viewed as an *implicational* hierarchy. To that end we construct data-driven transitivity hierarchies of two-place verb meanings based on the Valency Patterns Leipzig (ValPaL) database using Guttman scaling. We look at how well the hierarchies conform to strict scalarity (one-dimensionality) and, through matrix randomization, test whether their strengths are significant. We then go on to construct slightly different hierarchies based on simple counts of instances of two-participant coding frames for a given verb meaning across languages, rather than through the Guttman scaling procedure, which yields less resolution and is not designed for missing data. Finally, we assess whether the members of the hierarchies fall into semantic verb classes. The concluding section summarizes the results.

Keywords: transitivity hierarchy of two-place verb meanings, implicational hierarchies, Guttman scaling, matrix randomization, semantic verb classes, Valency Patterns Leipzig database

1 Introduction

Implicational hierarchies, a basic tool of linguistic typology since the 1960s and 1970s (cf. Greenberg 1963; Berlin and Kay 1969; Silverstein 1976; Keenan and Comrie 1977), have lately been partly challenged on various grounds (cf. Cysouw 2003; but also the ensuing discussion: Dryer 2003; Plank 2003). One of the main questions that have been raised is whether the putative implicational nature of these hierarchies has to be understood in absolute terms or in statistical terms

***Corresponding author: Gontzal Aldai,** University of the Basque Country / Euskal Herriko Unibertsitatea, Unibertsitateko ibilbidea 5, 01006 Vitoria-Gasteiz, Araba, Spain,
E-mail: gontzal.aldai@gmail.com
Søren Wichmann, University of Leiden & Kazan Federal University, P.N. van Eyckhof 3, 2311 BV Leiden, The Netherlands, E-mail: wichmannsoeren@gmail.com

only. If one takes the latter approach it is clear that some innovations are needed regarding statistical methods for the specific task at hand. A major first attempt at checking the statistical strength of linguistic implicational hierarchies is Wichmann (2016). This was followed up by the introduction of a significance test in Wichmann (2015) (written after the 2016 publication).

In the present paper, based on a further development of the methodology of Wichmann (2015), we attempt to test the "hierarchy of two-place predicates"[1] (Tsunoda 2015) – also called the "verb-type hierarchy" (Tsunoda 1981), the "transitivity scale of two-place predicates" (Tsunoda 1985), the "Transitivity Hierarchy" (Malchukov 2015), or the "ranking of transitivity-prominence" (Haspelmath 2015). Our aim is to investigate whether there are solid statistical grounds for maintaining some previously proposed version of that hierarchy or a similar one. The data we use for testing the hierarchy are from the database of the Valency Patterns Leipzig (ValPaL) project (Hartmann et al. 2013), available online at http://valpal.info/.

The paper is organized as follows. Section 2 briefly introduces the hierarchy at issue and its different versions. Section 3 presents our goals and methodology and discusses some of the difficulties relating to this endeavor. Section 4 provides results of Guttman scaling and tests for significance. It also provides rankings of two-place verb meanings in the database based on counts of instances of two-participant coding frames for a given verb meaning, and discusses the degree to which ranks are isomorphic with semantic classes. Section 5 summarizes and offers perspectives for further research.

2 The hierarchy of two-place verb meanings

2.1 Tsunoda's hierarchy

As is well known, the "hierarchy of two-place predicates" or the "transitivity scale of two-place predicates", first introduced by Tsunoda (1981) as the "verb-type hierarchy", is a seminal attempt to capture the notion of prototypical

[1] Tsunoda (2015) uses the term 'predicates' to refer to what we will be calling 'verb meanings', i.e. cross-linguistic comparison meanings which are very often expressed as verbs in most languages but which may occasionally also appear as adjectives or even nouns. Verb meanings will be written in small capitals in the present paper. The terms 'verb' and (infrequently) 'predicate' will be used here only to refer to items of specific languages; the former when it is an actual verb, the latter more generally. As for the term 'two-place' (alternatively, 'two-participant'), see discussion in Section 3.3.

transitivity in relation to semantic verb classes. Based on a small cross-linguistic study of two-place verb meanings (mainly from ergative languages), Tsunoda proposed a hierarchy of semantic verb classes where the prototypical transitive verbs (those which most typically take a transitive case frame) were at the left end of the scale, and “as we go down the scale, transitive case frames are less likely to occur” (Tsunoda 1985: 390). Although Tsunoda, as far as we know, never described his hierarchy using the word “implicational”, it is usually interpreted in this way (e.g. Levin 2006: 6; Haspelmath 2015: 142). The most recent version of the hierarchy (Tsunoda 1985 and Tsunoda 2015) is reproduced in (1).

(1) Tsunoda’s “transitivity scale of two-place predicates” (Tsunoda 1985: 388)
1. *Direct effect on patient* > 2. *Perception* > 3. *Pursuit* > 4. *Knowledge* > 5. *Feeling* > 6. *Relationship* > 7. *Ability*

Tsunoda (1985) also introduced two subtypes within two of his verb classes: the first class of ‘direct effect on patient’ was further divided into the subtypes 1A (resultative) and 1B (non-resultative); and the second class of ‘perception’ was split into the subtype 2A (‘patient more attained’) and the subtype 2B (‘patient less attained’). The ‘predicates’ (‘verb meanings’ in our terminology; see note 2) offered by Tsunoda as illustrative of each of the classes distinguished are given in (2).

(2) Tsunoda’s exemplar ‘predicates’ for each semantic class (Tsunoda 1985: 388)
1A: KILL, BREAK, BEND
1B: HIT, SHOOT, KICK, (EAT)[2]
2A: SEE, HEAR, FIND
2B: LISTEN, LOOK
3: SEARCH, WAIT, AWAIT
4: KNOW, UNDERSTAND, REMEMBER, FORGET
5: LOVE, LIKE, WANT, NEED, FOND, FEAR, AFRAID, ANGRY, PROUD, BOAST
6: POSSESS, HAVE, LACK, LACKING, RESEMBLE, SIMILAR, CORRESPOND, CONSIST
7: CAPABLE, PROFICIENT, GOOD

Tsunoda’s hierarchy has been appreciated as a fundamental contribution. Yet, a number of problems have also been pointed out (cf. Malchukov 2005: 75–77 for a

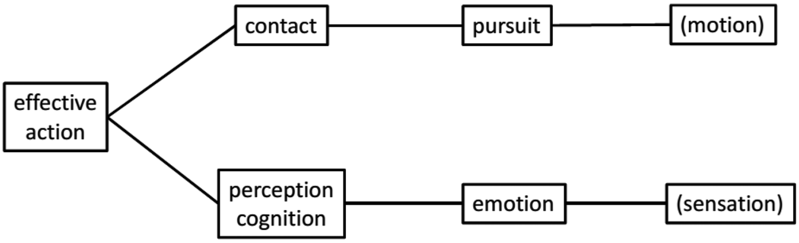
[2] The verb-meaning EAT does not appear in Tsunoda (2015: 1576), i.e. the final revision of Tsunoda (1985: 388).

summary). There has been some doubt about the implicational character of the hierarchy (cf. Lazard 1998: 60). In addition, it would seem that the division into subtypes within two classes of the scale disrupts the general hierarchy and complicates its testing. Perhaps most importantly, several scholars have pointed out that Tsunoda's hierarchy actually conflates different semantic dimensions (cf. Lehmann 1991: 234).

2.2 Malchukov's (2005) amendment

Malchukov (2005) proposed to revise Tsunoda's hierarchy by dividing it into two independent sub-hierarchies. The new version, which he called the "two-dimensional verb type hierarchy" is given in (3).

(3) The "two-dimensional verb type hierarchy" of Malchukov (2005: 81)



2.3 Haspelmath's (2015) testing

Based on the cross-linguistic database of the ValPaL project, Haspelmath (2015) arrived at a data-driven transitivity hierarchy of verb meanings that is cross-linguistically representative. He based his hierarchy on the concept of "transitivity prominence" of verb meanings, i.e. the percentage of transitively encoded verbs among all counterpart verbs, where transitive encoding is defined as using the same coding frame as that of the verb BREAK in the language at hand. Additionally, he compared his transitivity hierarchy with Tsunoda's and Malchukov's proposals, which were based on a more restricted set of data as well as on intuition. His conclusion was the following (2015: 144):

> My cross-linguistic study of transitivity prominence has largely confirmed the earlier studies by Tsunoda (1985) and Malchukov (2005) [...]. While these studies were formulated in terms of implicational scales, I studied transitivity prominence purely quantitatively and

found decreasing transitivity prominence in the series ‘break’, ‘hit’, ‘see’, ‘search for’, ‘know’, ‘like’ and ‘look at’.

3 Goals and methodology

3.1 General goals

The present paper has five main goals and steps. In a first step, we set out to construct a data-driven matrix of transitive coding for two-place verb meanings. Like Haspelmath (2015), we take our data from the ValPaL database. Once we have created a transitivity matrix of verb meanings that goes at least some way towards being cross-linguistically representative (but see Sections 3.2 and 3.7 below for some concerns even with these data), we will be able to produce and test new hierarchies of transitivity.

Secondly, besides constructing a matrix of transitive coding, we want to also take a look at the non-transitive two-place codings in the ValPaL database, in order to see whether hierarchies for these other coding frames can be established. To that end, we propose three operational non-transitive coding classes (see Section 3.5), and create one matrix for each of them. These three new matrices should help us to improve our understanding of the behavior of non-transitive two-place codings.

The third step of this paper, and a major motivation behind it, is to test statistically whether the matrices of two-place verb meanings constructed from the ValPaL significantly abide by the criterion of scalarity. Scalarity of a dataset is the same as its degree of one-dimensionality, which is the extent to which the dataset conforms to an implicational hierarchy. Rather than hoping for the hierarchies to be exceptionless, as in the mindset of some typologists, we actually expect to encounter exceptions. Nevertheless, after measuring the amount of exceptions we may still see strong hierarchies, something which requires a significance test to make sure that the patterns represented by these hierarchies are real and not due to chance. In carrying out the statistical testing, we follow the methodology in Wichmann (2015) but with some crucial improvements regarding how missing data are handled. These improvements allow for investing more confidence in the significance test used in this paper than in the one used in Wichmann (2015). Our approach is described in Section 3.6 below.

Fourthly, in addition to testing our four matrices for scalarity, we also want to construct four hierarchies based on simple counts of instances and percentages, rather than through the less transparent scaling procedure. As mentioned in Section 2.3 above, this task has been previously carried out by Haspelmath (2015), but only for the case of transitive coding and not for non-transitive codings. Thus, while we do not expect to add much regarding the transitive-coding hierarchy, the results for the other three hierarchies will represent completely novel contributions.

Finally, since the verb hierarchies proposed in the literature (at least until the studies based on the ValPaL data) were grounded on semantic verb *classes* (see (1) and (3) above), we want to test whether a hierarchy would be arranged on the basis of verb classes rather than just being a line-up of individual verbs with no discernable associations between specific semantic patterns and different sections of the hierarchy. The same goal is also pursued by Haspelmath (2015: 142–144). However, Haspelmath mainly focuses on prototypical members of each semantic class.

3.2 The ValPaL data

The data used here represent a subset of the ValPaL database (Hartmann et al. 2013: http://valpal.info) selected for the purpose of investigating two-participant predicates (see Section 3.3 for discussion), normally verbs. The ValPaL database provides data on valency patterns for a selected sample of 80 verb meanings from 36 languages around the world.[3] This project represents a tremendous effort towards increasing our understanding of basic coding frames (i.e. basic case-marking and agreement; for these terms, see Section 3.4) and of alternations, for different verb meanings and across languages. The authors acknowledge that "our 3[6] languages do not give a satisfactory picture of world-wide diversity" and that "in the case of verb meanings we do not know how to even address the issue of representativeness, other than by intuition" (Haspelmath 2015: 134). Yet, on the whole, the ValPaL database constitutes a crucial improvement regarding data on valency patterns and, moreover, is an extremely helpful tool for cross-linguistic comparison.

In this study, we have used data on basic coding frames only, and our contribution is thus complementary to those of Wichmann (2015, 2016), who

[3] Ainu, Balinese, Bezhta, Bora, Chintang, Eastern Armenian, Emai, English, Even, Evenki, German, Hokkaido Japanese, Hoocąk, Icelandic, Italian, Jakarta Indonesian, Jaminjung, Japanese (standard), Ket, Korean, Mandarin Chinese, Mandinka, Mapudungun, Mitsukaido Japanese, Modern Standard Arabic, Nen, N||ng, Ojibwe, Russian, Sliammon, Sri Lanka Malay, Xârâcùù, Yaqui, Yorùbá, Yucatec Maya, Zenzontepec Chatino.

looked at implicational verb hierarchies as derived from data on alternations and did not take into account basic coding frames. Here we draw upon information regarding basic two-place uses of verbs, selecting data from the database on verbs that are typically coded with two participants. In other words, instances of verbs that in a given language take as their basic coding one participant have been excluded from this study, and those taking three participants have been included only when the third participant is optional. In addition, we have applied the criterion that every verb meaning to be considered should be represented in at least half of the languages of the sample (for discussion on missing data, see Section 3.5 below). This selection procedure resulted in a list of 38 verb meanings which, within the limitations of the ValPal dataset, can be considered the most typical two-place verb meanings across languages. Thus, tables consisting of data on the 38 selected verb meanings across the 36 languages represented constitute the database used for this study, both for constructing new data-driven hierarchies of two-place verb meanings and for statistical tests.[4]

3.3 Terminology: "two-place" and "bivalent"

The notions of "two-place" and "bivalent" as used of a verb instantiating a given verb meaning are important to this work, but not entirely unproblematical. The term "bivalent" is obviously related to "valency". Thus, for using the former term we would need a clear definition of the latter, which would imply a clear-cut distinction between arguments and adjuncts. Unfortunately, as many linguists (including the ValPaL editors) acknowledge, we do not have good definitions of "valency" or "argument" that would be valid for all cases. Haspelmath and Hartmann (2015: 50) argue that "locational phrases and instrument nominals are hard to classify uniquely as arguments or adjuncts. [...] Thus, for quite a few cases we did not have a unique way of distinguishing between arguments and adjuncts, and the ValPaL database is therefore not consistent in this regard. [...] As a result, the number of arguments is not a kind of information that should be taken as important for cross-linguistic comparison" (see further contributions to Wichmann 2014 concerning problems with the argument / adjunct distinction).

Consequently, in this work we use the term "two-place" (alternatively "two-participant") instead of "bivalent" for the verbs instantiating the verb meanings under examination. The term "two-place" (employed by Tsunoda among others)

[4] These tables along with the software developed for this paper can be accessed at https://github.com/Sokiwi/Guttman.

is meant here to be neutral regarding the analysis of the verbal frame in terms of arguments and adjuncts. In other words, a two-place verb in a given language is one which is typically framed with two participants, regardless of whether those participants should be analyzed as arguments or adjuncts. Therefore, although it seems true that the number of arguments may not be taken as "important for cross-linguistic comparison" (Haspelmath and Hartmann 2015: 50), the ValPaL database proves that (either based on intuition or on frequency of use) the number of participants is a variable that can be employed for comparison across languages, as we try to do in the present paper. See, nevertheless, more discussion on the issue of "three-place verbs" in Section 4.2 below.

3.4 Coding classes: Initial practical assumptions and examples

The ValPaL data on basic coding frames are language-specific. Therefore, out of all the language-specific coding frames, we have proposed a few clusters and have arrived at four cross-linguistic coding classes, which we consider to be representative of coding frames across languages (see more on this issue in Section 3.5). The first step in the process of arriving at the four classes is to propose a working definition of "transitive coding" which can be applied to each of the languages in the database and which can also be used cross-linguistically, as a "comparative concept". Our working definition of transitive coding is the one in Haspelmath (2015): A given verb in a given language is coded as transitive if it has two participants that are coded like the two main arguments of the prototypical transitive verb BREAK. As is customary, we can label the two arguments in a transitive coding frame A and P (from Agent and Patient, respectively). A and P are formal coding types which can readily be identified in each language according to the specific coding devices of that given language. This identification process is illustrated below through a few examples.

The main coding devices in any language are, following the terminology adopted in the ValPaL project, "flagging resources" (case or adpositional marking) and "indexing resources" (bound person marking on the verb). To these, "ordering resources" can be added, although word order is not as productive (as the main coding device) across languages. Thus, identifying which verbs have and do not have transitive coding in a given language is straightforward in most cases.

The simplest case is one in which a language makes general use of flagging resources. This is illustrated by the Icelandic examples in (4–7).

(4) Icelandic (Barðdal 2013)
Strákur-inn *braut* *rúðu-na* *með* *steini*
boy-the.NOM broke.NOM3SG glass-the.ACC with stone.DAT
"The boy broke the window with a stone."

A prototypical transitive verb, e.g. *brjóta* 'break' in (4), codes its two main participants with the nominative (NOM) and accusative (ACC) cases, which correspond respectively to the micro-roles of "breaker" (A argument) and "broken thing" (P argument). The coding frame schema that the ValPaL presents for *brjóta* 'break' in Icelandic is the following: 1-NOM V.agr[1] 2-ACC (með+3-DAT). (Note that in addition there is nominative agreement – indexing – on the verb. Note also that the third participant, corresponding to the "breaking instrument" micro-role, appears in parentheses, indicating optionality). Consequently, the verb *elta* 'follow' in (5), which shows the same frame as *brjóta* 'break', also has transitive coding.

(5) Icelandic (Barðdal 2013)
Drengir-nir *eltu* *stelpur-nar*
boys-the.NOM followed.NOM3PL girls-the.ACC
"The boys followed the girls."

However, the verb *fylgja* 'follow, accompany' in (6) does not have transitive coding, because it does not show the same frame as *brjóta* 'break'.

(6) Icelandic (Barðdal 2013)
Drengir-nir *fylgdu* *stelpu-num*
boys-the.NOM followed.NOM3PL girls-the.DAT
"The boys accompanied the girls."

A coding that is not transitive either but differs from the pattern in (6) is shown in (7).

(7) Icelandic (Barðdal 2013)
Strák-num *líkaði* *nýja* *leikfang-ið* *sitt*
boy-the.DAT liked.NOM3SG new.NOM toy-the.NOM his.NOM
"The boy liked his new toy."

The difference between the coding frames in (6) and (7) is that in the former the participant with the micro-role of "follower" (a proto-agent) is coded in the NOM case and the participant with the micro-role of "followee" (a proto-patient) is coded as DAT, while in (7) the “liker” (a proto-agent) is coded as DAT and the “liked entity” (a proto-patient) is coded in the NOM case. (See more discussion on our coding frame classes in 3.5 below).

If a given language does not make use of flagging resources for marking its core participants, the identification of coding frames is based on indexing resources (and possibly on ordering resources). This is illustrated with the Yucatec Maya head-marking examples in (8–9), where CMPL stands for “completive status”, and D2 is a deictic enclitic.

(8) YUCATEC MAYA (Lehmann 2013)
T-u *pa’-ah-Ø* *kalomk’iin* *le* *xibpal*
PRFV-SBJ3 break-CMPL-OBJ3 window DEM boy
yéetel *tuunich-o’*
with stone-D2
“The boy broke the window with a stone.”

The prototypical transitive verb *pa’* ‘break’ in (8) encodes its two main participants by means of indexing (agreement in the verbal complex). The coding frame schema presented in the ValPaL data for the Yucatec Maya *pa’* ‘break’ verb is the following: Sbj[1].V.obj[2] 2 1 (yéetel+3). Note that, as in Icelandic, the third participant, corresponding to the “breaking instrument” micro-role, appears in parenthesis, indicating optionality. From the pattern of the verb *pa’* ‘break’, then, we can see that the verb *w-áant* ‘help’ in (9), for instance, also has transitive coding.

(9) YUCATEC MAYA (Lehmann 2013)
T-in *wáant-ah-Ø* *le* *xibpal-o’b-o’*
PRFV-SBJ1SG help-CMPL-OBJ3 DEM boy-PL-D2
“I helped the boys.”

Notice that the third participant in (8) is coded by means of a prepositional flagging resource. Indeed, all the languages in the database without flagging of core participants have at least some flagging resource for the marking of non-core participants. (See more on the coding of head-marking languages in Section 3.5).

3.5 Coding classes: Non-transitive coding

In an attempt to go beyond the study of transitive coding, we have also examined the non-transitive basic coding frames of the two-place verb meanings in the database. On the basis of the ValPal data at hand and also spurred by previous proposals (cf. Malchukov 2005), we have grouped all the language-specific non-transitive coding frames in three coding classes or macro-classes (each with its associated matrix). Thus, all in all, we have arrived at a four-way classification of two-place verb meanings. To arrive there, it was necessary to take some operational (partly subjective) decisions and to create new comparative concepts. Together with the transitive coding-class, which we have termed 1–2 (canonical coding), we found evidence for creating not just two new classes which depart from transitive coding, 1–3 (oblique-object coding) and 3–1 (inverted coding[5]) in our terminology, but also a third class termed 1–3LOC (locative-object coding).[6] In what follows we discuss the rationale behind this four-way classification – especially the one behind the distinction between 1–3 and 1–3LOC – and the practical implementations of this procedure.

A central assumption behind our classification is that we may create a comparative concept around the dative (DAT) case (cf. Haspelmath 2003: 213). Ideally, we want to make this cross-linguistic DAT-like coding correlate with our label 3. Thus, in a 1–3 coding frame, the proto-patient participant is coded in a DAT-like way; and in a 3–1 coding frame it is the proto-agent participant which is coded in a DAT-like way. The ValPaL two-place examples where the proto-agent participant is non-canonically coded are few and in most cases can readily be assimilated to a 3–1 pattern. Because of the scarcity of examples, we have been as generous as possible when accepting a given coding frame into the 3–1 class: any non-canonical encoding of the proto-agent was sufficient for a verb to be assigned to 3–1.[7] However, the two-place examples where it is the other participant which is non-canonically coded are much more numerous and very heterogenous. This is the rationale for establishing a 1–3LOC class. We distinguished between coding devices that could be assimilated to DAT, assigning the

[5] For this term and the notion underlying it, see Bossong (1998).

[6] The approach of Say (2014) to the definition of coding classes bears some similarities to ours. In cases of non-canonical two-place coding classes this author looks for the element carrying the non-canonical case marker, the so-called “locus of non-transitivity”, and classifies the coding frame according to whether it is the semantic agent or patient which is encoded in a non-canonical way. Since only a few examples are presented, we cannot compare our approaches in detail. On a general level, it deserves mentioning that Say (2014) does not deal with instances that would belong in our 1–3LOC class.

[7] The following are the specific coding frames that we included in our inverted coding class (3–1): DAT-NOM in German, Icelandic, Japanese (standard) and perhaps in Ket; experiencer (EXP)-NOM and EXP-ACC in Mitsukaido Japanese; *a* + NP-NOM in Italian; lative (LAT)-ABS in Bezhta; and DAT-ACC in Sri Lanka Malay. To these we added a few NOM-NOM frames (see the text).

corresponding verbs to the 1–3 class, and coding devices that could not be assimilated to DAT. The latter were then included in the 1–3LOC class. A guiding principle was that cases of 'recipient' + 'motion to' received the 3 label, whereas cases of 'location at' + 'motion from' received the 3LOC label.

Throughout the whole procedure, we have crucially relied on the basic two-participant coding frames as presented in the ValPaL project for each language examined. If a given language had a DAT case (labeled as such in the ValPaL database),[8] then a two-participant basic coding frame of the type 1-DAT was included in our 1–3 (oblique-object) macro-class. Yet, if for whatever reasons, a given language did not have a DAT case in the database, then we looked for coding frames with case-markers or adpositions that could be assimilated to datives, chiefly including allatives (motion to),[9] in order to also assign them to the oblique-object macro-class.

Thus, the ValPaL data have been particularly helpful for discriminating between what we have called "obliques" and "locatives". In most instances, the basic two-place coding frames that each language shows in the ValPaL database can be assigned to one of our four macro-classes with relative ease. In many cases, the ValPaL data give a language-specific label "loc" or "LOC" to refer to either one particular case-marker or adposition (loc), or more generally to various case-markers or adpositions of locative meaning (LOC).[10] In all those instances, following the ValPaL data, we have included the corresponding verbs in our 1–3LOC macro-class. In addition, language-particular case-markers or adpositions which specifically show static location or ablative (motion from) meanings have also been included in our 1–3LOC macro-class. Conversely, case-markers or adpositions that convey allative or a similar meaning (provided they are not included in the database with a LOC label) have been assigned to our 1–3 macro-class. (Other, more uncommon, instances of case-markers or adpositions

[8] The following languages show a DAT case in the ValPaL database: Even, Evenki, German, Hokkaido Japanese, Icelandic, Jaminjung, Japanese (standard), Ket, Korean, Mitsukaido Japanese, Nen, N||ng, Russian, and Sri Lanka Malay. That is, 14 languages out of 36.

[9] Other grammatical means of conveying DAT meaning readily come to mind; e.g. serial verb constructions and applicatives. These are rare in the database, however, and in any event they do not disturb our strategy of looking for case-markers or adpositions that can be assimilated to DAT.

[10] The following languages show a LOC or loc label in the ValPaL database: Ainu, Bezhta, Bora (loc/instr), Chintang, Eastern Armenian, English, Even, Evenki, German, Hoocak, Icelandic, Italian, Jakarta Indonesian, Jaminjung, Japanese (standard), Ket, Korean, Mandinka, Mitsukaido Japanese (loc/com), Modern Standard Arabic, Nen, Ojibwe, Russian, Sri Lanka Malay, Xârâcùù, Yaqui, Yoruba, Yucatec Maya, and Zenzontepec Chatino. This makes for a total of 29 languages out of 36.

were considered on an individual basis. Most instrumentals and genitives were assigned to the 1–3 macro-class). We have found a few instances where this practice may be somewhat doubtful. Nevertheless, we have followed the ValPaL data.

The Ket language (Yeniseic, Siberia) can serve to illustrate the above practical implementation, as well as our decisions regarding head-marking languages. This language has clause-level head-marking for arguments (i.e. indexing, verb agreement) and case-markers and postpositions for non-arguments. The ValPaL data for the basic two-participant coding frames in Ket show the following picture. Besides the transitive class (1–2), with indexing on the verb for subjects and objects, which has twenty verb meanings in it, eight more basic two-participant coding frames appear, all of them with only between one and four verb meanings. We left aside the two coding frames containing only the verbs for MEET and SING, because they show a rather idiosyncratic behavior (reciprocal and unergative, respectively). Thus, we had six basic coding frames to classify and proceeded to make decisions on macro-class assignment on a case-by-case basis. The first of these coding frames (containing the verbs for FEAR, LEAVE and SMELL) shows an ABL case-marked second participant. The second coding frame (with GO, LIVE, SIT and SIT DOWN) has a second participant labeled LOC in the database. The verbs in these two coding frames were assigned to the 1–3LOC macro-class. The third coding frame (with FOLLOW and KNOW) has two unmarked participants, only the first of which is indexed on the verb. The fourth frame (SEARCH FOR and THINK) has a second participant in the adessive case. The fifth frame (LIKE[11] and SHOUT AT) has a second participant in the DAT case. Finally, the sixth frame (LOOK AT) has a second participant with the postposition *towards*. The verbs in these four coding frames were assigned to the 1–3 macro-class.[12]

Ket can also illustrate one of the issues that the above practice may present. Within the label LOC in the ValPaL data of two-participant coding frames for Ket, there is actually more than one case-marker. One of them (the one used with

[11] According to the micro-roles and the examples given for LIKE in Ket, it seems to be the first participant that appears in the DAT case. Thus, it seems that this verb should actually belong to the inverted (3–1) class.

[12] It is difficult to formulate general principles for macro-class assignment covering all constructions in all languages, especially when it comes to the kinds of idiosyncrasies that Ket is so rich in. Thus, different researchers might sometimes make different choices. An anonymous reviewer noted that s/he would prefer to assign the fourth frame, where the second participant is in the adessive case, to the 1–3LOC macro-class since, according to Vajda (2004: 27), adessive case in Ket denotes a static location with respect to (mostly animate) reference points. This is a legitimate alternative to our 1–3 assignment; perhaps even a better choice.

LIVE, SIT) conveys static location. Another one (used with GO, SIT DOWN) expresses motion to. The problem that arises here is that the latter marker is formally identical to the DAT marker. At any rate, since the ValPaL database distinguishes two coding frames, one labeled 1-LOC and the other 1-DAT, we have also distinguished between the two, and assigned the former to our 1–3LOC macro-class and the latter to our 1–3.

The ValPaL data for English (a language more accessible to most readers) also illustrate our procedure nicely, and show some of its possible shortcomings. Besides the transitive (1–2) coding frame (to which 19 verb meanings pertain), the English data show five more two-place coding frames. The first of these (for APPEAR, CLIMB, FALL, FEEL PAIN, GO, LIVE, ROLL, and SIT) is labeled 1-LOC. Thus, we included the verbs having this coding frame within our 1–3LOC macro-class. The coding frames where the second participant is marked with the prepositions *about* (THINK) and *of* (BE AFRAID) did not show clear criteria for being assigned to either 1–3 or 1–3LOC.[13] Nevertheless, we also decided to include them in our 1–3LOC macro-class. In contrast, the coding frames where the second participant is marked with the prepositions *at* (LOOK AT, SHOUT AT) and *to* (LISTEN TO) were assigned to our 1–3 macro-class. This choice was somewhat arbitrary, but since the ValPaL database distinguishes between coding frames labeled 1-LOC, 1-*at*, and 1-*to*, we chose to retain some of the discrimination by assigning the 1-LOC cases to our 1–3LOC macro-class and the 1-*at* and 1-*to* cases to our 1–3 macro-class. Needless to say, different choices of collapsing the three-way distinction into a two-way distinction were available.

Finally, there are two difficult cases which may be worth mentioning, even if they are represented by just a handful of examples, namely the few instances of NOM-NOM and ABS-ABS coding frames. The former shows up in just 4 languages in the database (Hokkaido Japanese, Standard Japanese, Korean, Mitsukaido Japanese) and only in relation to four verb meanings (FEAR, FEEL PAIN, HEAR, LIKE). The latter is found in just 2 languages (Jaminjung, Nen) and in connection with four verb meanings (COOK, EAT, KNOW, SHAVE). For convenience, the NOM-NOM coding frame has been included in the inverted (3–1) macro-class, and the ABS-ABS frame in the oblique-object (1–3) macro-class.

To sum up, our four-way classification of two-place verb meanings distinguishes: a) a transitive-coding class (1–2); b) an oblique-object class (1–3), where the proto-patient participant is coded in an oblique way, typically including the DAT case; c) a locative-object class (1–3LOC), where the proto-patient participant is typically coded as LOC; and d) an inverted class (3–1), where it is the proto-

[13] Note that these cases constitute a minority of all the cases at hand. Also, the most significant verb meanings in the literature are seldom coded by means of minor coding frames.

agent, rather than the proto-patient, that takes a non-canonical (often DAT) marking.[14]

3.6 Methodology for statistical testing

As already stated, for the statistical testing of the degree of scalarity of our hierarchies we followed Wichmann's (2015) methodology, based on Guttman scaling. From the table of 38 verb meanings and 36 languages, four matrices (with 1's and 0's) were constructed, one for each of the basic coding frame classes presented above (which were called 1–2, 1–3, 1–3LOC and 3–1, respectively):

(a) 1–2: NOM-ACC, ERG-ABS, etc. (transitive coding)
(b) 1–3: NOM-DAT, NOM-OBL, ERG-OBL, ABS-OBL, etc. (oblique-object coding)
(c) 1–3LOC: NOM-LOC, ERG-LOC, ABS-LOC, etc. (locative-object coding)
(d) 3–1: DAT-NOM, OBL-NOM, OBL-ABS, etc. (inverted coding).

Following the Guttman scaling procedure the rows and columns of each matrix with verb meanings as columns and languages as rows is rearranged in a so-called "scalogram" where the cells with 1's are gathered towards one corner of the matrix (cf. Wichmann 2016 for an illustration and more explanation). This is achieved by counting the number of 1's in each row and column and arranging the rows and columns such that the sums of 1's consistently decrease in both of the two dimensions when moving away from the corner where the 1's are gathered. Ideally there should now not be any 0's interspersed among the 1's or the other way around. If that is the case, the Guttman Coefficient (GC) is 100%. This coefficient decreases with the number of deviations from complete scalarity. Each misplaced 1 or 0 counts as an error. The algorithm used seeks to minimize the number of errors while maximizing the number of 1's considered to not be errors. For instance, given a sequence such as 1 0 1 0 0 0 1 0, the algorithm takes steps from left to right, counting the number of 1's included and the number of errors. Stopping at the first position will include a single 1 and there will be

[14] As an anonymous reviewer points out, the four-way classification of coding classes is founded on the combination of two different principles: meaning and formal coding. First, the identification of the transitive class is based on verb meanings that are believed to constitute the core of (semantic) transitivity. Once the transitive class is formally identified within each language, the identification of the other three macro-patterns is based on language-specific coding devices.

two errors – the 1's in position three and seven. Stopping at position two will again include one 1 and there will now be three errors – the 0 in position two and the 1's in positions three and seven. The optimum in this case is to stop at position three, counting two errors – the 0 in position two and the 1 in position seven. In a sequence such as 0 0 1 0 0 0 0 0 stopping at position one will result in no 1's included and two errors – the 0 in position one and the 1 in position three; stopping at position two will again result in no 1's included and three errors – the two 0's in positions one and two and the 1 in position 3; stopping at position three will result in one 1 included and two errors – the 0's in position one and two; this last choice is then the optimum. In this process missing cells (NA's) are simply ignored. For the purpose of rank-ordering verb meanings missing cells are also ignored. For instance, in a sequence such as 1 1 NA 1 0 1 0 0 the algorithm chooses position six as the optimum, with one error – the 0 in position five and four 1's included. The verb meaning would then receive a score of 4 for the purpose of ranking. It is not clear whether this is the best choice of dealing with missing data for the purpose of rank-ordering, but it is at least simple. To calculate the Guttman Coefficient the total number of errors is divided by the number of filled cells, expressing this as a percentage and finally subtracting it from 100% (Wichmann 2015: 157). The GC is thus a simple first measure of scalarity. A scalogram can be constructed relatively easily by hand, but we have a revised version of the program in Wichmann (2015), provided along with this paper,[15] which secures replicability. This program also computes a GC and outputs a scale, in this case of verb meanings.

Next, a significance test applying matrix randomization (cf. Janssen et al. 2006) was applied, following Wichmann (2015: 159), with some improvement. As in Wichmann (2015), we randomize the matrix 9,999 times without changing the sums of 1's in each row and column, compute the GC for each randomized version, and let the proportion of cases where the GC is equal to or higher than GC of the original matrix represent our *p*-value. Thus, the *p*-value is the probability of obtaining the given GC (or higher) by chance. Conventionally, $p < 0.05$ is considered significant.

For randomization the function permatfull of R's vegan package is used (Oksanen et al. 2011). A problem here is that this function cannot be applied in cases of missing values (empty cells). Empty cells occur when a specific verb is lacking in a given language, when a verb is basically coded as one-place or as trivalent (three-place) in a specific language, or for other reasons.

[15] R functions written for this paper are available at https://github.com/Sokiwi/Guttman.

In order to make randomization possible the empty cells must be filled (imputed) with 0's and 1's in some sensible way. Although recognizing that this was a far from adequate solution, the take on the problem of Wichmann (2015: 162) was to make two matrices where all empty cells were filled by respectively 0's and 1's, run the significance test on both, and then make the assumption that if the GC was significant in both cases it should also be significant for the original matrix with missing cells. Here we improve on the method of imputation as follows. The basic idea is to impute 0's and 1's such that the new matrix has a GC as close as possible to the original one (GCorig). The R program written for this goes to each missing cell, first puts in a 0 and then a 1 and choses the value that results in a GC closest to GCorig. This is first done columnwise and then rowwise. The matrix with the GC closest to GCorig is then selected for further refinement. In our experience, the GC will already differ from GCorig by less than 0.1% at this point. Nevertheless, the matrix is further improved by randomly selecting a cell that was previously empty, changing the 0 to 1 or the other way around, accepting the edit if the difference between the new GC and GCorig diminishes or discarding it when the difference does not diminish. This is tried until the new GC and GCorig are identical or until a large number of trials, N, have been carried out leaving little hope that the imputed matrix will ever get exactly the same GC as GCorig. We used $N=10{,}000$. In the end the differences between GCs for imputed matrices and GCorig ranged between 0 and 0.03. Finally, the imputed matrices were submitted to the significance test.

3.7 Some difficulties

It is clear that a task like the present one is fraught with difficulties. Although we cannot devote too much space here to discussing all the obstacles we have encountered, it is worth mentioning that cross-linguistic comparability of the verb meanings under examination is one of the most important issues in this undertaking (see Haspelmath 2010; for a general discussion and Haspelmath and Hartmann 2015; for the solutions adopted in the ValPaL project). Another problem, which was already mentioned, relates to the making of the database, including data elicitation. Finally, as reflected in the previous subsection, the statistical test becomes somewhat idiosyncratic in the situation where there are missing data. Nevertheless, we believe that the decisions adopted in this paper, all of them based on the typological literature on the topic and particularly on the ValPaL project, represent reasonable solutions to the various obstacles mentioned. See more discussion on these issues in Section 5.

4 Results

4.1 Results of Guttman scaling and significance testing

Each of the four data matrices was turned into a scalogram and rankings were obtained as described in Section 3.6 above. These rankings are provided in (10–13). A greater-than sign separates each rank. Ties (verb meanings having the same rank) are separated by commas and are given in random order.

(10) Guttman scale for 1–2 (transitive coding)
BREAK, CUT, EAT, KILL > FRIGHTEN > BEAT, HIT, WASH > GRIND, SEE > COOK, HUG > KNOW > PEEL, TOUCH, SMELL > SEARCH FOR, HEAR > BUILD, COVER, DIG, FOLLOW, LIKE > SHAVE, HELP > FILL, LOOK AT > MEET, FEAR > DRESS > LEAVE > SHOUT AT > THINK > CLIMB > GO > SIT DOWN, SIT, LIVE

(11) Guttman scale for 1–3 (oblique-object coding)
SHOUT AT > FOLLOW, HELP > LOOK AT > GO > FEAR, CLIMB, SIT DOWN, LIVE, HUG > LIKE, LEAVE, HEAR, SIT, TOUCH > THINK, SEARCH FOR, EAT, MEET, KNOW, DRESS, SHAVE, SMELL, WASH, BEAT, BREAK, BUILD, COOK, COVER, CUT, DIG, FILL, FRIGHTEN, GRIND, HIT, KILL, PEEL, SEE

(12) Guttman scale for 1–3LOC (locative-object coding)
SIT, LIVE > SIT DOWN > GO > LEAVE > CLIMB > THINK > FEAR, SMELL > DIG, FILL, SHOUT AT, HEAR, SEARCH FOR > MEET, LIKE, SHAVE, FOLLOW, EAT, BEAT, BREAK, BUILD, COOK, COVER, CUT, DRESS, FRIGHTEN, GRIND, HELP, HIT, HUG, KILL, KNOW, LOOK AT, PEEL, SEE, TOUCH, WASH

(13) Guttman scale for 3–1 (inverted coding)
LIKE > HEAR > SEE > KNOW > MEET, FEAR, LOOK AT > BEAT, BREAK, BUILD, CLIMB, COOK, COVER, CUT, DIG, DRESS, EAT, FILL, FOLLOW, FRIGHTEN, GO, GRIND, HELP, HIT, HUG, KILL, LEAVE, LIVE, PEEL, SEARCH FOR, SHAVE, SHOUT AT, SIT, SIT DOWN, SMELL, THINK, TOUCH, WASH

The results of applying the statistical methodology explained in Section 3.6 to the hierarchies obtained through Guttman scaling are displayed in (14).

(14) Guttman Coefficients and significance tests of the hierarchies in (10) to (13).

GC is the Guttman Coefficient of each of the original matrices, GC(imp) that of imputed matrices, and *p*(imp) the *p*-value of GC(imp). The proportion of empty cells is in all cases 9.8%.[16]

Coding	**GC**	**GC(imp)**	***p*(imp)**
1–2 (transitive)	94.41	94.44	**0.0001**
1–3 (oblique-object)	94.57	94.59	**0.0002**
1–3LOC (locative-object)	95.54	95.54	**0.0001**
3–1 (inverted)	99.27	99.27	0.4967

The Guttman Coefficient (GC) values in (14) are high, always close to or above 95 for the four hierarchies. This says something about the strength of the hierarchies interpreted as implicational ones – conventionally, GC = 85 is considered the cut-off for what can reasonably be considered a candidate for an implicational hierarchy. Nevertheless, one should be cautious in interpreting the numbers. As observed by Wichmann (2015: 159), "the more skewedness there is in the proportion between 1's and 0's – whether one or the other dominates – the easier it is to get high GC values." Ignoring empty cells, the percentage of 1's and 0's is, respectively, 72.1% vs 27.9% for 1–2; 7.0% vs 93.0% for 1–3; 12.4% vs 87.6% for 1–3LOC; and 1.3% vs 98.7% for 3–1. The high GC's, then, cannot be taken as conclusively supporting a hierarchy as being implicational – here it is more interesting to look at the results of the significance test.

Values of *p*(imp) below 0.05 indicate that the GC's are not likely to have been gotten by chance. There is support here for 1–2, 1–3, and 1–3LOC codings, but not for 3–1 codings. The lack of support for 3–1 is not surprising given that this is a rare coding frame. Ignoring this last type of coding, there is good support for the hypothesis that there are implicational hierarchies of verb meanings across languages. On the other hand, Guttman scaling did not lead to very resolved hierarchies. All four of them have few ranks and many ties. Moreover, creating hierarchies based on this technique gets somewhat convoluted and potentially controversial when there are missing data-points. For these reasons we leave Guttman scaling behind and move on to formulate hierarchies based on slightly different principles leading to more resolution and better transparency.

[16] The percentage of empty cells (9.8%), for which no data are available, refers to the case where three-place instances of verbs are discarded (N/A) unless one participant is clearly optional (see Sections 3.3 and 4.2).

4.2 Results regarding the transitivity hierarchy of two-place verb meanings

Considering also the data on the 38 selected verb meanings in the database that typically are two-place verbs and focusing on transitive coding as in (10) above, we now construct a slightly different hierarchy based on simple counts of instances rather than through the Guttman scaling procedure. In this way we arrive at the hierarchy in (15), which is ordered from the verb meaning that has the highest percentage of transitive coding frames in our 36 languages (out of the total of basic two-place coding frames for that verb meaning) to the one that has the lowest percentage.

We encountered several issues when producing this hierarchy. One of our major concerns when calculating the ranking was how to deal with three-place verbs. In principle, our criterion was to include three-place instances of verbs only when the third participant is optional, so that the two mandatory participants can be said to constitute the basic bivalent (two-place) frame. (Trivalent verbs, therefore, were excluded). Yet it was not always easy to tell from the ValPal data when this stipulation is met. For this reason we have calculated two rankings: one disregarding three-place instances of verbs (unless one participant is clearly optional), and the other taking them into account. As can be observed from the data, if three-place verbs are discarded, a number of verb meanings (i.e. BEAT, COVER, HIT, and PEEL) get a 100% transitive coding, which they would otherwise not have. In addition, there are six more verb meanings for which we have not been able to find any non-transitive basic coding frame. Thus, we have a total of ten 100% transitive-coding verbs (when three-place verbs are removed). In order to also rank these, we have considered (and given in parentheses) the absolute number of transitive coding among the 36 languages under discussion.[17]

Another difficulty when dealing with the data may be illustrated with the verb-meaning EAT. This verb meaning shows transitive coding (1–2) for all the 36 languages in the database. However, in a few instances in the database, there are particular verb meanings that have more than one type of basic coding frame

[17] Therefore, when more than one figure is given, a) the first one makes reference to the percentage of transitive coding (NOM-ACC, etc.) of a given verb meaning out of the total of basic *two-place* coding frames for that verb meaning; b) the figure in parentheses gives the *absolute* number of two-place transitive coding frames that a given verb meaning has in the 36 languages under consideration; and c) the last figure, when different from the first, provides the percentage of transitive coding of a given verb meaning out of the total of basic coding-frames (*two- or three-place*) for that verb meaning. (This last value, incidentally, is the hardest to determine).

in a given language. This happens twice with the verb-meaning EAT. In Jaminjung (northern Australia), there are two predicates for EAT: one is simplex (*ganimindany*), the other is complex (*thawaya gagba*). The first one has transitive coding (ERG-ABS), while the second one has ABS-ABS coding. In Sliammon (Salish, Pacific North America) there are also two verbs for EAT. The former (*mək*^{w}*-t*) shows transitive coding (subject and object agreement on the verb); the latter (*ʔiɬtən*) has 1–3 coding (subject agreement and an oblique NP for the 'eaten food' micro-role). Therefore, in order to calculate the transitivity ranking in these cases, we had to make additional decisions. We proceeded as follows: we counted a total of 38 instances of verbs for EAT (36 + 2) and then divided the 36 transitive codings by the total of 38 instances. This produced a transitivity ranking of 0.947 for EAT.

For the purpose of the scalogram-derived hierarchies above a given verb meaning for a given coding frame will either score a 1 or a 0. If there is at least one synonym which scores a 1 in all languages, as in the case of EAT, the verb meaning will rank on the top of the hierarchy. For this reason some verb meanings may appear *higher* up in the Guttman scales in (10–13) than in the more fine-grained hierarchies in (15–18). Conversely, some verb meaning may appear *lower* in the Guttman scales. This happens when there is missing data which, as mentioned earlier (Section 3.6), are counted as absences of positive scores. For instance, for the 1–2 coding scheme data are missing from one language for FRIGHTEN, from three languages for GRIND, and from eight languages for BUILD. Therefore FRIGHTEN, GRIND, and BUILD are on successively lower levels of the Guttman scale in (10) although they are all indicated as having 100% transitive encoding in (15). A less simplistic approach to the handling of missing data for the purpose of Guttman scaling could have been applied, but since we have the alternative ranking represented by (15–18) we did not go beyond the simplistic version of Guttman scaling.

(15) ValPaL two-place verb meanings ranked for transitive coding (1–2)

1. BREAK	1.00 (36)	20. KNOW	.886 (31)
2. CUT	1.00 (36)	21. SMELL	.882 (30)
3. KILL	1.00 (36)	22. SEARCH FOR	.882 (30)
4. FRIGHTEN	1.00 (35)	23. TOUCH	.833 (30)
5. HIT	1.00 (34) / .971	24. HEAR	.806 (29)
6. BEAT	1.00 (34) / .944	25. MEET	.781 (25)
7. GRIND	1.00 (33)	26. HELP	.757 (28)
8. PEEL	1.00 (31) / .886	27. FOLLOW	.744 (29)
9. BUILD	1.00 (28)	28. LIKE	.725 (29)
10. COVER	1.00 (28) / .875	29. LOOK AT	.722 (26)

11. WASH	.972 (35)	30. FEAR	.605 (23)
12. COOK	.970 (32)	31. THINK	.529 (18)
13. FILL	.964 (27) / .917	32. LEAVE	.514 (19)
14. SHAVE	.964 (27)	33. SHOUT AT	.500 (16)
15. EAT	.947 (36)	34. CLIMB	.500 (19)
16. DRESS	.940 (19) / .788	35. GO	.065 (2)
17. DIG	.935 (29)	36. SIT DOWN	.036 (1)
18. SEE	.917 (33)	37. SIT	.032 (1)
19. HUG	.914 (32)	38. LIVE	.031 (1)

4.2.1 A comparison with Haspelmath's (2015) results

The results regarding what may be considered our transitivity hierarchy of two-place verb meanings in (15), which obviously is ranked similarly to the Guttman scale we obtained in (10), are comparable to those obtained in Haspelmath (2015: 143). This is certainly expected, since Haspelmath's data and ours are taken from the same source: the ValPaL database. However, some differences may also be anticipated, since Haspelmath's goal is to determine the ranking of transitivity vs. *intransitivity* of the verb meanings at issue, whereas ours is to establish the ranking *of two-place verb meanings* regarding their coding as transitive vs. *other two-place* types of coding. In other words, Haspelmath builds his ranking on the percentage of transitively encoded verbs among *all* counterpart verbs, whereas we calculate our percentages (mostly) for two-place basic uses of verbs.[18] All in all, however, the two hierarchies are quite similar.

The hierarchy seems to divide into segments, each of which could be singled out for further study. There seems to be a top segment containing the meanings BREAK, CUT, KILL, FRIGHTEN, HIT, BEAT, GRIND, PEEL, BUILD, COVER, WASH, COOK, FILL, SHAVE, EAT, DRESS, DIG; a middle area would include SEE, HUG, KNOW, SMELL, SEARCH FOR, TOUCH, HEAR, MEET, HELP, FOLLOW, LIKE, LOOK AT, FEAR; finally it appears that there is a bottom region, divided in turn into two subparts: first THINK, LEAVE, SHOUT AT, CLIMB, and finally GO, SIT DOWN, SIT, and LIVE. Certainly, the most appealing segment to study further, but probably also the most challenging, would be the one in the middle region.

[18] It is perhaps worth stressing that this technical difference was not expected to bring about any substantial disparity with Haspelmath's results. As explained in Sections 3.2 and 3.3, we chose as our goal in this paper to examine only two-participant verb meanings.

4.3 Results regarding non-transitive coding of two-place verb meanings

The hierarchy in (15) provides a ranking of the ValPaL two-place verb meanings based on the percentage of transitive coding (NOM-ACC, ERG-ABS) in their basic form. Thus, for the verb meanings at the top of the hierarchy (BREAK, for instance) all or most of their basic coding frames are transitive, and this percentage diminishes as we go down the hierarchy. However, (15) does not say anything about the non-transitive basic coding frames that each of the verb meanings may take across languages.

The hierarchies in (16–18) show the ranking of the two-place verb meanings in the database based on the percentage of instances where they enter into, respectively, the oblique-object frame (16), the locative-object frame (17) and the inverted frame (18).

(16) ValPaL two-place verb meanings ranked for oblique-object coding (1–3)

1. SHOUT AT	.483	20. DRESS	.050
2. FOLLOW	.257	21. SHAVE	.037
3. LOOK AT	.250	22. SMELL	.031
4. HELP	.229	23. EAT	.028
5. GO	.193	24. WASH	.028
6. FEAR	.156	25. BEAT	0
7. THINK	.121	26. BREAK	0
8. SEARCH FOR	.118	27. BUILD	0
9. CLIMB	.114	28. COOK	0
10. TOUCH	.111	29. COVER	0
11. LIKE	.111	30. CUT	0
12. SIT DOWN	.107	30. DIG	0
13. LIVE	.094	31. FILL	0
14. LEAVE	.091	32. FRIGHTEN	0
15. HEAR	.088	33. GRIND	0
16. HUG	.086	34. HIT	0
17. MEET	.065	36. KILL	0
18. SIT	.065	37. PEEL	0
19. KNOW	.057	38. SEE	0

(17) ValPaL two-place verb meanings ranked for locative-object coding (1–3LOC)

1. SIT	.903	20. BEAT	0
2. LIVE	.875	21. BREAK	0
3. SIT DOWN	.857	22. BUILD	0

4. GO	.742	23. COOK	0
5. LEAVE	.455	24. COVER	0
6. CLIMB	.429	25. CUT	0
7. THINK	.333	26. DRESS	0
8. FEAR	.219	27. FRIGHTEN	0
9. MEET	.129	28. GRIND	0
10. SMELL	.094	29. HELP	0
11. DIG	.065	30. HIT	0
12. LIKE	.056	31. HUG	0
13. SHAVE	.037	32. KILL	0
14. FILL	.036	33. KNOW	0
15. SHOUT AT	.034	34. LOOK AT	0
16. HEAR	.029	35. PEEL	0
17. SEARCH FOR	.029	36. SEE	0
18. FOLLOW	.029	37. TOUCH	0
19. EAT	.028	38. WASH	0

The hierarchy in (16) ranks the verb meanings in the ValPaL database according to how often they are coded in an oblique-object frame in their basic form. The hierarchy in (17), in turn, ranks verb meanings according to how often they appear in the locative-object frame. As expected, both hierarchies have some similarities and both show some degree of inverse relationship with the hierarchy in (15). There are, however, also clear differences between (16) and (17). The top-most verb meanings in (16) are SHOUT AT, FOLLOW, LOOK AT, and HELP, which are not ranked particularly high in (17). The same may hold for the verb meanings SEARCH FOR and TOUCH. Conversely, the highest ranked verb meanings in (17), SIT, LIVE, and SIT DOWN, give verbs that usually select a static locative argument, but in specific languages may take a general oblique constituent, and are thus quite highly ranked in the hierarchy of (16) too. A similar picture obtains for verb meanings such as GO, LEAVE, and CLIMB. Furthermore, an important conclusion that may be drawn from (16), apart from the top ranking of verb meanings such as the aforementioned SHOUT AT, FOLLOW, LOOK AT, HELP, and SEARCH FOR, is the fact that almost half of the 38 two-place verb meanings in the selected database do not have any oblique-object basic coding in the 36 languages at issue. This is still clearer in the hierarchy of (17): only half of the verbs in the list have some locative-object basic coding.

Finally, the ranking of the two-place verb meanings in the database according to an inverted coding is given in the hierarchy of (18). This last hierarchy is even more restricted than the previous two. As a matter of fact, only seven verbs in the database have some inverted basic coding for any of the 36 languages under consideration.

(18) ValPaL two-place verb meanings ranked for inverted coding (3–1)

1. LIKE	.167	20. FRIGHTEN	0
2. HEAR	.118	21. GO	0
3. SEE	.088	22. GRIND	0
4. KNOW	.057	23. HELP	0
5. MEET	.032	24. HIT	0
6. FEAR	.031	25. HUG	0
7. LOOK AT	.028	26. KILL	0
8. BEAT	0	27. LEAVE	0
9. BREAK	0	28. LIVE	0
10. BUILD	0	29. PEEL	0
11. CLIMB	0	30. SEARCH FOR	0
12. COOK	0	31. SHAVE	0
13. COVER	0	32. SHOUT AT	0
14. CUT	0	33. SIT	0
15. DIG	0	34. SIT DOWN	0
16. DRESS	0	35. SMELL	0
17. EAT	0	36. THINK	0
18. FILL	0	37. TOUCH	0
19. FOLLOW	0	38. WASH	0

As can be observed from (18), the top-ranked verb meanings in the database according to an inverted basic coding are LIKE, HEAR, SEE, and KNOW. And only MEET, FEAR, and LOOK AT show any other inverted basic coding frame. Moreover, the percentages of inverted basic coding among all the 36 languages studied are low even for the highest ranked verbs (LIKE, HEAR, or SEE).

A first conclusion that may be drawn at this point is that, among the most typical two-place verb meanings, transitive basic coding (NOM-ACC, ERG-ABS) is by far the most widely used across languages and across verb meanings. As Haspelmath (2015: 139) puts it, "the prominence of transitivity does seem to be a robust language universal". Only seven verb meanings in the ValPaL dataset (and only nine of the languages) show any basic inverted coding. Similar observations can be made for the basic oblique-object coding.[19]

[19] A larger database with more verb meanings would probably be needed to encounter more cases that typically have 1–3 and 3–1 codings, with all the effort and time this would entail. The

4.4 Results regarding semantic verb classes

The two main proposals for verbal hierarchies in the typological literature (Tsunoda 1985; Malchukov 2005) are designed on the basis of semantic verb classes: see (1) and (3) above (cf. also Levin [2015: 1605–1607], for what she calls "the appeal of verb classes"). As mentioned earlier (cf. Sections 2.3 and 3.1), Haspelmath (2015) already checked Tsunoda's (1985) and Malchukov's (2005) verb classes against the ValPaL data. Actually, Haspelmath (2015: 142–144) only discusses the verb meanings BREAK, HIT, SEE, LOOK AT, SEARCH, KNOW, and LIKE, one for each of the first seven semantic classes in Tsunoda's proposal (see (2) above). However, there are more verb meanings in the ValPaL database that can be tested for their possible grouping into semantic classes. Specifically, the following verb meanings in Tsunoda's classification (grouped and ranked by semantic classes) appear in the ValPaL dataset:

(19) Tsunoda's verb meanings in ValPal (Tsunoda 2015: 1576)

1A (resultative direct effect):	KILL, BREAK
1B (non-resultative direct effect):	HIT
2A (attained perception):	SEE, HEAR
2B (less attained perception):	LOOK AT
3 (pursuit):	SEARCH
4 (knowledge):	KNOW
5 (feeling):	LIKE, FEAR
6, 7:	–

As for Malchukov's (2005) proposal, although he does not give a list of specific verb meanings for each of his semantic classes, the following would be the ones appearing in the ValPaL dataset:

(20) Malchukov's verb meanings in ValPal (Malchukov 2005: 81)

a) First sub-hierarchy:

1A (effective action):	BREAK
2A (contact):	HIT, TOUCH
3A (pursuit):	SEARCH FOR, FOLLOW
4A (motion):	GO

following verb meanings, among others, could show 1–3 coding: ASSIST, ATTACK, CALL, GRAB, HOLD ON TO, LISTEN TO, LOOK FOR, SHOOT AT, WAIT FOR, WAVE AT (Cooreman 1994: 60; Lazard 1998: 146). The following could have 3–1 coding: BECOME AFRAID OF, BECOME ANGRY ABOUT/WITH, BECOME GLAD ABOUT, BECOME SURPRISED BY, FORGET, LACK, LOSE, REMEMBER (Bossong 1998: 261).

b) Second sub-hierachy:

1B (effective action):	BREAK
2B (perception / cognition):	SEE, HEAR / KNOW
3B (emotion):	LIKE, FEAR
4B (sensation):	(BE HUNGRY, FEEL COLD)

Now, when we have more than one verb meaning in our database which, according to Tsunoda or Malchukov, would fall into the same class, we do not always get similar results. Thus, while BEAT and HIT, presumably both impact verbs (Tsunoda 1985; includes them among "non-resultative effect" verbs; and Malchukov 2005; among contact verbs) get very similar rankings in all four hierarchies and particularly in (15), pursuit verbs such as SEARCH FOR and FOLLOW are located rather distantly in this hierarchy (0.882 and 0.744, respectively). The same holds for the case of perception verbs such as SEE (0.917) and HEAR (0.806), or for emotion verbs such as LIKE (0.725) and FEAR (0.605).

All in all, regardless of the number of verb meanings considered, our results concerning Tsunoda's and Malchukov's semantic verb classes and their testing against the ValPaL data are very similar to Haspelmath's (2015) conclusions that "there is a broad match" between Tsunoda's hierarchy and the ValPaL data, but that "the position of LOOK (AT) is the most striking discrepancy" (Haspelmath 2015: 142). In addition, as also pointed out by Haspelmath (2015: 142) on the basis of his selected seven verb meanings, our results do not easily accommodate Tsunoda's verb classes either. Wichmann (2016) similarly found that a hierarchy based on ValPaL data for alternations did not show segments corresponding to easily defined semantic classes and warned that "one should avoid defining large groups such as 'effective action' or 'perception' a priori" (Wichmann 2016: 426).

As regards Malchukov's (2015) verb classes, on the whole, his proposal of a two-dimensional scale does fit better with our ranking in (15) than Tsunoda's unidimensional hierarchy. Thus, the difficulty with the relative ranking of the verbs SEARCH FOR and KNOW is solved in the two-dimensional scale, since each of these verbs now belongs to an independent dimension. This solution is, moreover, intuitively sound on semantic grounds, because it does appear that the verb meanings in (20a) vary along a dimension of patienthood or affectedness[20] of proto-patients, while those in (20b), experiential verbs, vary along a dimension of agentivity of proto-agents (Malchukov 2005: 80–81).

[20] "Affectedness" is a term which is not easy to define and may even be controversial. We mean to use it here in the most neutral way. For recent discussion on the concept of affectedness, see Næss (2004), Kittilä (2008), Beavers (2011), and Croft (2012).

Nevertheless, Malchukov's verb classes are not without problems either, as Malchukov himself points out. For instance, regarding the first sub-hierarchy, the verbs HIT and TOUCH would, according to Malchukov (2005: 83), belong in the semantic class of contact verbs, which would be ranked higher than pursuit verbs (SEARCH FOR, FOLLOW). In our ValPaL ranking of (15), however, TOUCH is placed below SEARCH FOR (and above FOLLOW). Additionally, concerning the second sub-hierarchy, the verb HEAR is ranked lower than KNOW, while Malchukov (2005: 111) argues that "cognition verbs are intermediate between the perception and emotion verbs"; although it is also true that Malchukov (2005: 81) groups perception and cognition verbs in the same class.

For these and other reasons, Malchukov (2005: 111–113) puts forward a further "comprehensive semantic map for transitive splits"; i.e. a new proposal of verb classes now based on the concept of two-dimensional semantic map (cf. Haspelmath 2003). Even with this new proposal, however, Malchukov acknowledges that "the map is still incomplete in that further semantic types can be introduced" (2005: 112). Indeed, the general impression is that the more verb meanings one takes into account the more difficult it becomes to come up with a comprehensive classification of verb classes which would fit the ranking arrived at.

Finally, there is one last issue which seems to be important concerning Malchukov's hypothesis of a two-dimensional scale. If a clear-cut division between the two dimensions put forward by Malchukov occurs, one can expect to find across languages that the verbs pertaining to each of the sub-hierarchies have, apart from the canonical transitive coding, a major type of coding of their own. That is, whenever the verbs in the first sub-hierarchy (varying along a dimension of patienthood or affectedness of proto-patients) depart from the canonical transitive coding, they would tend to take a coding frame different from the verbs in the second sub-hierarchy (varying along a dimension of agentivity of proto-agents). This prediction is, at least in part, confirmed by the data.

The hierarchy in (16), which ranks oblique-object coding, shows that, as predicted, the great majority of verbs at the top of the ranking arguably belong to Malchukov's sub-hierarchy along affectedness of proto-patients: SHOUT AT, FOLLOW, LOOK AT, HELP, GO, SEARCH FOR, CLIMB, TOUCH. Conversely, if we examine the hierarchy in (18), which ranks inverted coding, practically all of the top verbs in this ranking, again as expected, pertain to the sub-hierarchy along agentivity of proto-agents: LIKE, HEAR, SEE, KNOW, FEAR. These data, therefore, constitute important evidence in favor of a two-dimensional scale of two-place verb meanings.

There is, however, one proviso that needs to be pointed out. There are some verbs that clearly belong to the sub-hierarchy along agentivity of proto-agents which still are quite highly ranked in the hierarchy in (16): emotion verbs such as FEAR and LIKE are the best examples, but even HEAR could be mentioned. In contrast, practically no verb belonging to the sub-hierarchy along affectedness of proto-patients appears well ranked in the hierarchy in (18). As a matter of fact, most of the typical verbs here (FOLLOW, HELP, SEARCH FOR, TOUCH) do not have any instance of inverted coding in the database. Only LOOK AT can be taken as exceptional in a few instances; yet, this could be due to its semantic relationship with SEE.

In sum, the data substantiate the hypothesis of a two-dimensional scale of transitivity, with one sub-hierarchy for variation along affectedness of proto-patients and another sub-hierarchy for variation along agentivity of proto-agents (i.e. experiential verbs). While the great majority of verbs belonging to the sub-hierarchy of affectedness show the oblique-object frame as their main non-canonical coding, experiential verbs (belonging to the sub-hierarchy of agentivity) can take an inverted coding but also an oblique-object coding, besides the canonical transitive coding frame.[21]

5 Conclusions

In the previous sections we investigated the possibility of establishing a transitivity hierarchy of two-participant verb meanings on the basis of the ValPaL data. We now turn to a discussion of how those results should be interpreted. In general, the results do not stray too far from proposals concerning transitivity hierarchies in the literature, but there are various amendments to be made.

The established proposals in the typological literature on the topic are that (a) there are verb hierarchies that are implicational in nature; (b) these hierarchies are arranged in semantic verb classes; (c) regarding specifically the transitive hierarchy, there seem to be two main sub-hierarchies that need to be distinguished: one along affectedness of proto-patients, the other along agentivity of proto-agents.

[21] As noted by an anonymous reviewer, this asymmetry seems to stem from the very semantic rationale behind the two sub-hierarchies. This is so because, typically, verb meanings with low agentivity (LIKE, HEAR, SEE, KNOW, FEAR) simultaneously imply weakly affected objects. Yet, the opposite does not always hold.

All of these statements ought to be constrained, refined, and properly interpreted, as detailed in the following points.

1. Three of the four hierarchies, importantly including the basic transitive coding-class, passed the test for one-dimensionality. Thus, it seems that coding classes generally induce implicational scales among verb meanings. In contrast, Wichmann (2015), and Wichmann (2016) found more limited support for implicational hierarchies with respect to the selection of different alternations.
2. The transitivity hierarchy of two-place verb meanings arrived at in this paper to some extent resembles the main proposals in the literature (Tsunoda 1985; Malchukov 2005). However, there are also differences, especially concerning the middle part of the hierarchy, which incidentally is the most interesting to examine.
3. Although an implicational transitivity hierarchy can be supported, there would be numerous and serious uncertainties regarding the organization of the hierarchy into semantic verb classes. Malchukov's proposal of a two-dimensional scale is better suited than Tsunoda's hierarchy to account for the types of non-transitive coding which two-place verbs may take down the scale(s). Yet, even Malchukov's two hierarchies show a fair number of problems when it comes to accommodating semantic verb classes.
4. Malchukov's two-dimensional scale appears as a sound hypothesis and is consistent with the data; however, as we have seen above, not all the verbs in the sub-hierarchy along agentivity of proto-agents (experiential verbs) take the inverted pattern as the only non-canonical coding frame in all languages (cf. Bossong 1998). Some languages may use an oblique-object frame here too, even if the latter may seem more suited for the sub-hierarchy along affectedness of proto-patients. Therefore, it is not as easy to make predictions based on this sub-hierarchy as a totally independent two-dimensional scale would suggest.
5. Concerning the new hierarchies for intransitive rankings the most important findings are as follows. The verb meanings most often coded for oblique-object (1–3) are: SHOUT AT > FOLLOW > LOOK AT > HELP (for the full hierarchy see 16). These are verb meanings involving a weakly affected proto-patient. Although there is a tendency for them to take oblique-objects they can have transitive coding as well. The following verb meanings tend to have locative-object coding (1–3LOC): SIT > LIVE > SIT DOWN > GO > LEAVE > CLIMB (for the full hierarchy see 17). Verbs corresponding to these meanings are intransitive in most languages; nevertheless, they are typically two-place verbs. The following verb meanings have a tendency to take inverted coding (3–1): LIKE > HEAR > SEE > KNOW > FEAR (for the full hierarchy see 18, but note that

it should be taken with a grain of salt since the implicational nature could not be confirmed statistically). Verbs taking inverted coding are mostly experiential and involve a proto-agent having low agentivity. In most languages, however, they take transitive coding (1–2).

6. Much research is still needed on verb classes and semantic maps in order to disentangle these and other related issues. It could be the case, for instance, that more factors might intervene between verb classes as a whole and their transitivity ranking. Malchukov (2015: 87–89) mentions a number of other putative factors, but there might be even more variables involved: e.g. prototypicality of verb meanings within each semantic class, frequency of each verb, tense-aspect semantics, etc. One should not forget, however, that with every added parameter an explanatory model loses some of its power in the sense that it becomes easier to fit the model to any kind of situation.

Acknowledgements: SW's research was supported by an ERC Advanced Grant (MesAndLin(g)k, proj. no. 295918) and by a subsidy of the Russian Government to support the Programme of Competitive Development of Kazan Federal University.

References

Barðdal, Jóhanna. 2013. Icelandic valency patterns. In Iren Hartmann, Martin Haspelmath & Bradley Taylor (eds.), *Valency Patterns Leipzig*. Leipzig: Max Planck Institute for Evolutionary Anthropology. http://valpal.info/languages/icelandic (accessed 28 January 2017).

Beavers, John. 2011. On affectedness. *Natural Language & Linguistic Theory* 29. 335–370.

Berlin, Brent & Paul Kay. 1969. *Basic color terms: Their universality and evolution*. Berkeley: University of California Press.

Bossong, Georg. 1998. Le marquage de l'expérient dans les langues d'Europe. In Jack Feuillet (ed.), *Actance et valence dans les langues de l'Europe*, 259–294. Berlin: Mouton de Gruyter.

Cooreman, Ann. 1994. A functional typology of antipassives. In Barbara Fox & Paul J. Hopper (eds.), *Voice: Form and function*, 49–88. Amsterdam: John Benjamins.

Croft, William. 2012. *Verbs: Aspect and causal structure*. Oxford: Oxford University Press.

Cysouw, Michael. 2003. Against implicational universals. *Linguistic Typology* 7. 89–101.

Dryer, Matthew. 2003. Significant and non-significant implicational universals. *Linguistic Typology* 7. 108–128.

Greenberg, Joseph H. 1963. Some universals of grammar with particular reference to the order of meaningful elements. In Joseph H. Greenberg (ed.), *Universals of language*, 40–70. Cambridge, MA: MIT Press.

Hartmann, Iren, Martin Haspelmath & Bradley Taylor (eds.). 2013. *Valency Patterns Leipzig*. Leipzig: Max Planck Institute for Evolutionary Anthropology. http://valpal.info.

Haspelmath, Martin. 2003. The geometry of grammatical meaning: Semantic maps and cross-linguistic comparison. In Michael Tomasello (ed.), *The new psychology of language, vol. 2*, 211–242. Mahwah, NJ: Lawrence Erlbaum Associates.

Haspelmath, Martin. 2010. Comparative concepts and descriptive categories in cross-linguistic studies. *Language* 86. 663–687.

Haspelmath, Martin. 2015. Transitivity prominence. In Andrej Malchukov & Bernard Comrie (eds.), *Valency classes in the world's languages*, 131–147. Berlin: De Gruyter Mouton.

Haspelmath, Martin & Iren Hartmann. 2015. Comparing verbal valency across languages. In Andrej Malchukov & Bernard Comrie (eds.), *Valency classes in the world's languages*, 41–71. Berlin: De Gruyter Mouton.

Janssen, Dirk, Balthasar Bickel & Fernando Zuñiga. 2006. Randomization tests in language typology. *Linguistic Typology* 10. 419–455.

Keenan, Edward & Bernard Comrie. 1977. Noun phrase accessibility and universal grammar. *Linguistic Inquiry* 8. 63–99.

Kittilä, Seppo. 2008. Animacy effects on differential Goal marking. *Linguistic Typology* 12. 245–268.

Lazard, Gilbert. 1998. *Actancy*. Berlin: Mouton de Gruyter.

Lehmann, Christian. 1991. Predicate classes and participation. In Hansjakob Seiler & Waldfried Premper (eds.), *Partizipation: Das sprachliche Erfassen von Sachverhalten*, 183–239. Tübingen: G. Narr.

Lehmann, Christian. 2013. Yucatec Maya valency patterns. In Iren Hartmann, Martin Haspelmath & Bradley Taylor (eds.), *Valency Patterns Leipzig*. Leipzig: Max Planck Institute for Evolutionary Anthropology. http://valpal.info/languages/icelandic (accessed 28 January 2017).

Levin, Beth. 2006. Lexical semantics and argument realization VI. More on semantic determinants of argument realization: Evidence from transitivity. Handout for a DGfS/GLOW Summer School course, Stuttgart, August 2006. https://web.stanford.edu/~bclevin/dgfs06semdet.pdf (accessed 20 January 2017).

Levin, Beth. 2015. Verb classes within and across languages. In Andrej Malchukov & Bernard Comrie (eds.), *Valency classes in the world's languages*, 1605–1647. Berlin: De Gruyter Mouton.

Malchukov, Andrej. 2005. Case pattern splits, verb types and construction competition. In Mengistu Amberber & Helen de Hoop (eds.), *Competition and variation in natural languages: The case for case*, 73–118. Amsterdam: Elsevier.

Malchukov, Andrej. 2015. Valency classes and alternations: Parameters of variation. In Andrej Malchukov & Bernard Comrie (eds.), *Valency classes in the world's languages*, 73–130. Berlin: De Gruyter Mouton.

Næss, Åshild. 2004. What markedness marks: The markedness problem with direct objects. *Lingua* 114. 1186–1212.

Oksanen, Jari, F. Guillaume Blanchet, Roeland Kindt, Pierre Legendre, Peter R. Minchin, R. B. O'Hara, Gavin L. Simpson, Peter Solymos, M. Henry H. Stevens & Helene Wagner. 2011. vegan: Community ecology package. R package version 2.0–2. http://CRAN.R-project.org/package=vegan.

Plank, Frans. 2003. There's more than one way to make sense of one-way implications. *Linguistic Typology* 7. 128–140.

Say, Sergey. 2014. Bivalent verb classes in the languages of Europe: A quantitative typological study. *Language Dynamics and Change* 4. 116–166.

Silverstein, Michael. 1976. Hierarchy of features and ergativity. In R. M. W. Dixon (ed.), *Grammatical categories in Australian languages*, 112–171. Canberra: Australian Institute of Aboriginal Studies.

Tsunoda, Tasaku. 1981. Split case-marking in verb-types and tense/aspect/mood. *Linguistics* 19. 389–438.

Tsunoda, Tasaku. 1985. Remarks on transitivity. *Journal of Linguistics* 21. 385–396.

Tsunoda, Tasaku. 2015. The hierarchy of two-place predicates: Its limitations. In Andrej Malchukov & Bernard Comrie (eds.), *Valency classes in the world's languages*, 1575–1603. Berlin: De Gruyter Mouton.

Vajda, Edward J. 2004. *Ket*. Munich: LINCOM Europa.

Wichmann, Søren (ed.). 2014. Arguments and adjuncts cross-linguistically. [Special issue]. *Linguistic Discovery* 12(2). 1–84.

Wichmann, Søren. 2015. Statistical observations on implicational (verb) hierarchies. In Andrej Malchukov & Bernard Comrie (eds.), *Valency classes in the world's languages*, 155–181. Berlin: De Gruyter Mouton.

Wichmann, Søren. 2016. Quantitative tests of implicational verb hierarchies. In Taro Kageyama & Wesley M Jacobsen (eds.), *Transitivity and valency alternations: Studies on Japanese and beyond*, 423–444. Berlin: De Gruyter Mouton.