An incremental approach to gapping in Japanese

Petter Haugereid
Western Norway University of Applied Sciences

Proceedings of the 26th International Conference on
Head-Driven Phrase Structure Grammar

University of Bucharest

Stefan Müller, Petya Osenova (Editors)

2019

Stanford, CA: CSLI Publications

pages 42–57

Keywords: HPSG, Gapping, Japanese, Incremental Parsing

Haugereid, Petter. 2019. An incremental approach to gapping in Japanese. In Stefan Müller & Petya Osenova (eds.), *Proceedings of the 26th International Conference on Head-Driven Phrase Structure Grammar, University of Bucharest*, 42–57. Stanford, CA: CSLI Publications. DOI: 10.21248/hpsg.2019.3.

Abstract

Gapping in Japanese, which is an SOV language, differs from gapping in SVO languages in that the conjuncts with the elided verbs appear in non-final position. In this paper I present an incremental approach to gapping in Japanese, where it is assumed that an argument structure type is constructed in the non-final clause(s) in the gapping construction. This type is unified with the construction type created by the final clause resulting in identical construction types for all conjuncts in the construction.

1 Introduction

Gapping is a phenomenon that poses a challenge to lexicalist approaches given the fact that the main verb of one or more of the conjuncts in these constructions is elided. Example (1) (from Sag *et al.* (1985)) shows the prototypical gapping construction with a transitive sentence in the first conjunct, and two arguments, but no verb, in the second conjunct.

(1) Kim likes Sandy, and Lee Leslie.

The constituents of the conjunct with the elided verb, *Lee* and *Leslie*, are referred to as the *remnants*, and the constituents that have their roles in the conjunct with the verb, *Kim* and *Sandy*, are referred to as *correlates*.

The examples in (2)–(6) demonstrate gapping in Japanese (from Kato (2006, p. 1–14)). (2) is a conjunction of two transitive sentences where the verb of the first conjunct is elided. (3) shows that the elided verb cannot be in the second conjunct, as in English. (4) shows that gapping also may occur with intransitive verbs. (5) shows that there may be four dependents in each conjunct, and (6) shows that there may be more than one conjunct with a gap.

(2)

John-ga	hon-o	sosite	Mary-ga	hana-o	katta.
John-NOM	book-ACC	and	Mary-NOM	flower-ACC	bought

'John bought books, and Mary flowers.'

(3)

*	John-ga	hon-o	katta	sosite	Mary-ga	hana-o.
	John-NOM	book-ACC	bought	and	Mary-NOM	flower-ACC

(4)

John-ga	kayobi-ni	sosite	Mary-ga	doyoubi-ni	hasiru.
John-NOM	Tuesday-ON	and	Mary-NOM	Saturday-ON	run

'John runs on Tuesdays, and Mary on Saturdays.'

(5)

John-ga	kinou	Fred-ni	hon-o	sosite	Mary-ga	kyou
John-NOM	yesterday	Fred-DAT	book-ACC	and	Mary-NOM	today

Susan-ni	hana-o	katta.
Susan-DAT	flower-ACC	bought

[†] I would like to thank three anonymous reviewers and the audience at the HPSG 2019 conference in Bucharest, Romania, for very useful comments and suggestions.

'John bought books for Fred yesterday, and Mary flowers for Susan today.'

(6)

John-ga	hon-o	sosite	Mary-ga	hana-o	sosite	Fred-ga
John-NOM	book-ACC	and	Mary-NOM	flower-ACC	and	Fred-NOM

pen-o	sosite	Sue-ga	kitte-o	katta.
pen-ACC	and	Sue-NOM	stamp-ACC	bought

'John bought books, Mary flowers, Fred pens, and Sue stamps.'

According to Ross (1970), gapping operates forward in SVO languages like English. This is referred to as forward gapping (see (7)). And in SOV languages like Japanese, the verb appears in the last conjunct in gapping constructions (Ross, 1970). This is referred to as backward gapping (see (8)).

(7) a. SVO + SVO + SVO + ... + SVO ⇒
b. SVO + SO + SO + ... + SO

(8) a. SOV + SOV + SOV + ... + SOV ⇒
b. SO + SO + SO + ... + SOV

Gapping in Japanese is sometimes equaled to Right Node Raising (Kato, 2006, p. 55). Yatabe and Tanigawa (2018) claim that Japanese does not have gapping, only Right Node Raising. They base their argument on the fact that the apparent ellipsis only is at the right node of the conjunct, illustrated in (9), where it appears that the whole right node *nani o kau to yakusoku shita no* is gapped. According to Yatabe and Tanigawa (2018), the reading of (10), where the verb *kau* and the complementizer *to* are not gapped, should be the same as the reading of (9) if Japanese had gapping, but this reading is not available, and they present this as evidence that Japanese does not have gapping.

(9)

[Masao wa]	ashita,	(soshite)	[Hanako wa]	asatte
[Masao TOP]	tomorrow	(and)	[Hanako TOP]	day after tomorrow

[nani o]	kau	to	yakusoku	shita	no?
[what ACC]	buy-PRES	COMP	promise	do-PAST	NML

'What has Masao promised to buy tomorrow, and what has Hanako promised to buy the day after tomorrow?'

(10) ?*

[Masao wa]	ashita	kau	to,	(soshite)	[Hanako wa]
[Masao TOP]	tomorrow	buy-PRES	COMP	(and)	[Hanako TOP]

asatte	[nani o]	kau	to	yakusoku	shita
day after tomorrow	[what ACC]	buy-PRES	COMP	promise	do-PAST

no?
NML

'Same as (9)'

In this paper, I will not discuss whether or not gapping exists in Japanese. The aim will be to present an analysis of the examples in (2)–(6), where a verb is shared by two (or more) conjuncts. However, in the following I will refer to this phenomenon as gapping.

Gapping is a widely discussed phenomenon in the linguistic literature, and it is one of the hardest phenomena to handle in a grammar implementation. The analyses of gapping rarely find their way into grammar implementations. In this paper, the focus will be on implementability of accounts of gapping, and hence the perspective will be different from other, more theoretical, approaches. The hope is that it can complement the other approaches and show a way forward to how analyses of gapping can be implemented. Guided by limitations imposed by concerns about implementability and parser efficiency,[1] the account I will present is limited in scope, and only accounts for a fraction of the data on gapping found in the literature.[2]

2 Gapping in HPSG

In lexicalist theories, the syntactic structure is built up around heads which carry detailed information about the structure that will be built around them. This makes gapping constructions hard to account for, given that the verb, which is the head of the sentence, is missing.

Most HPSG approaches to gapping makes use of the linearization approach (Kathol, 1995; Beavers and Sag, 2004; Chaves, 2005; Crysmann, 2008; Kim and Cho, 2012). In this approach, the feature DOM(ain) (Reape, 1994) represents the linear order of phonological items, and this order is allowed to be different from the order in the constituent tree. This separation of linear order and constituent tree is powerful, and although relational constraints may be added to the grammar in order to impose restrictions on the order of the phonological items, it may put a heavy burden on the parser if it is not properly constrained.

Abeillé *et al.* (2014) present an alternative, construction-based HPSG approach to gapping. It is based on Mouret (2006), and does not make use of linearization. Instead it assumes that the constituents in the conjuncts with the elided verb, the *remnants*, form a non-headed constituent where the *synsem*s of the remnants are entered onto a CLUSTER list in HEAD. This constituent undergoes a unary rule *head-fragment-ph*. This *head-fragment* rule checks the HEAD values of the remnants (via the CLUSTER list) against the HEAD values of the correlates, which the rule accesses via the context SAL(ient)-(sub)UTT(erance) feature (SAL-UTT) (see (12)).

[1] The analysis presented is possible to implement with the LKB system (Copestake, 2002).

[2] More complex examples of gapping, for example including chains of control verbs as in (11) (from Sag *et al.* (1985)) and examples like (9), will be topic for future research.

(11) Pat wanted to try to go to Berne, and Chris to Rome.

(12) Syntactic constraints on *head-fragment-ph* (Abeillé *et al.*, 2014)

$$\textit{head-fragment-ph} \Rightarrow \begin{bmatrix} \text{CONTEXT} \mid \text{SAL-UTT} \left\langle \begin{bmatrix} \text{HEAD} \ \boxed{H_1} \\ \text{MAJOR} \ + \end{bmatrix}, \ldots, \begin{bmatrix} \text{HEAD} \ \boxed{H_n} \\ \text{MAJOR} \ + \end{bmatrix} \right\rangle \\ \text{CATEGORY} \mid \text{HEAD} \mid \text{CLUSTER} \left\langle \begin{bmatrix} \text{HEAD} \ \boxed{H_1} \\ \text{MAJOR} \ + \end{bmatrix}, \ldots, \begin{bmatrix} \text{HEAD} \ \boxed{H_n} \\ \text{MAJOR} \ + \end{bmatrix} \right\rangle \end{bmatrix}$$

In example (1), repeated here as (13), the correlates are *Kim* and *Sandy*. Consequently, their *synsem*s can be accessed via the SAL-UTT feature. The *head-fragment* rule checks that their head values match with those of the remnants, *Lee* and *Leslie*. In this way, the subcategorization frames of the conjuncts with elided verbs are guaranteed to correspond to the subcategorization frames of the conjunct with the verb.

(13) Kim likes Sandy, and Lee Leslie.

The tree in Figure 1 is an illustration of how the features SAL-UTT and CLUSTER account for the matching of the argument frames of the initial conjunct and a conjunct with an elided verb.[3]

In addition to the syntactic constraint shown in (12), there is a separate constraint on the *head-fragment* rule that assigns the semantic predicate that was assigned to the correlates, to the remnants. In (13), this means that the semantics of the second conjunct is *like'(lee',leslie')*.

There are some challenges to the approach to gapping in Abeillé *et al.* (2014), and in particular the syntactic constraints on *head-fragment-ph* shown in (12). While it is possible to match the HEAD values of the synsems on the CLUSTER list with those on the SAL-UTT list, one needs to know the length of the SAL-UTT and CLUSTER lists, unless one introduces some extra functionality for list matching. If there are two correlates and two remnants, as in (13), one needs a *head-fragment-ph* type that has SAL-UTT and CLUSTER lists of length two, and which matches the HEAD values of the two first items and the HEAD values of the two second items. If there are three correlates and three remnants, one needs another *head-fragment-ph* type with lists of length three, and so on. In addition, if one allows the matching elements on the lists to come in different order, the number of matching rules required becomes large.

A more serious problem with the *head-fragment* rule is how to make the items on the SAL-UTT and CLUSTER lists accessible to the rule at the same time. The access to the correlates on the SAL-UTT list in the *head-fragment* rule presupposes that the conjunct with the correlates has been parsed when the *head-fragment* rule is applied, and that the the correlates has been put on a SAL-UTT list. This list will have to be made accessible to the coordination rule, which pushes it down into the *head-fragment* rule, via the *head-comps* rule, as shown in Figure 1. The fact

[3]The analysis presented in Abeillé *et al.* (2014) is more elaborate than the illustration shown here. It includes a functionality that allows them to match constituents with differing HEAD values, like *adv* and *prep*, and *noun* and *adj*.

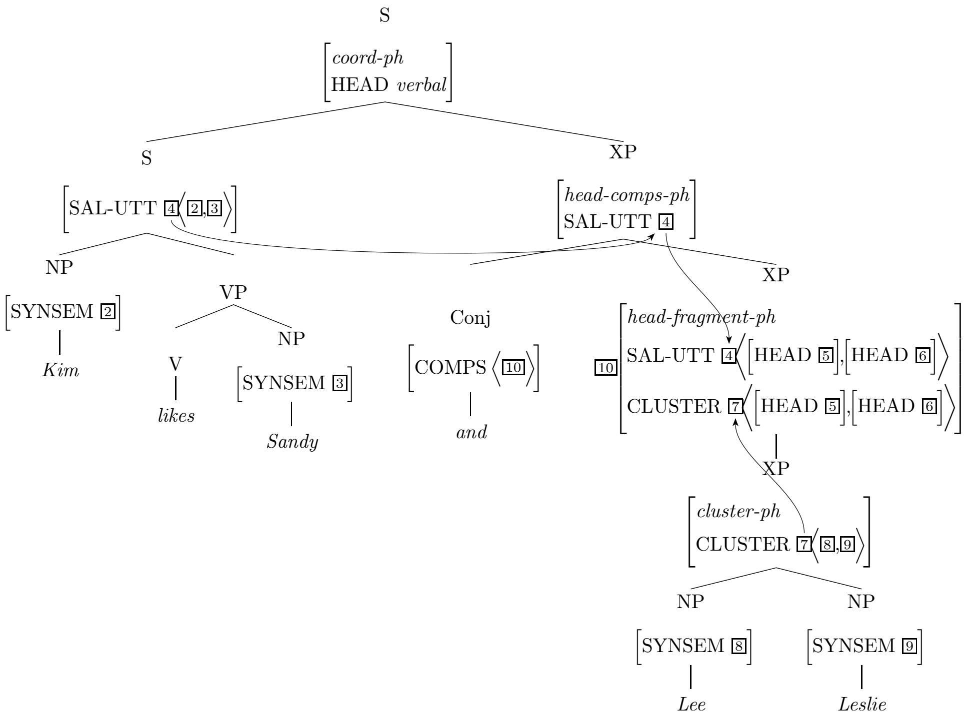


Figure 1: Simplified tree for (13) demonstrating the SAL-UTT and CLUSTER lists

that the CLUSTER list comes from below, and the SAL-UTT list comes from above, means that only one of the lists will be populated when the parser attempts to apply the *head-fragment* rule, irrespective of whether the parsing strategy is bottom-up or top-down. One of the lists will be empty until the whole coordination is parsed. This will lead to a large number of contexts where the *head-fragment* rules would be applicable, before both the lists are populated and the matching of the lists can be attempted, and it will lead to a massive burden of the parser.

Both the syntactic and semantic constraints assumed on the *head-fragment* rule in Abeillé *et al.* (2014) assume access to information in the conjunct with the verb. This makes sense in languages with forward gapping, like English, French and Romanian, but in a language like Japanese, where the remnants come before the correlates, one would have to wait for the final verb before the constraints required by the *head-fragment* rule would be made available. If one assumes a parser that works right-to-left, this can be accounted for, but it would be hard to defend from a psycholinguistic point of view.

In this paper, the incremental left-to-right approach to gapping in Haugereid (2017) will be adapted, and it will be shown how the left-to-right approach used to account for forward gapping also can be used for backward gapping, even though the verb only appears in the final conjunct. This is made possible given that the grammar is designed in such a way that a clause in principle can be parsed without a verb. The argument structure is assumed to originate from the syntactic rules, and the verb is treated as a kind of obligatory modifier. If there is no verb, the parse will result in an underspecified construction type which only reflects the argument structure of the clause, but not the predicate of the main verb.

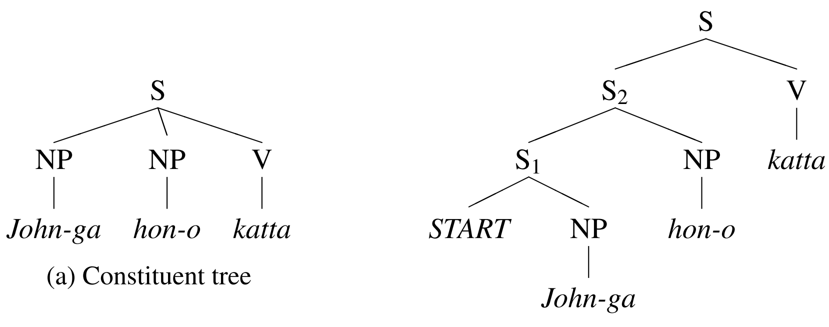
3 Analysis of gapping in Japanese

In Haugereid (2017), gapping in Norwegian, which is an SVO language, is accounted for by assuming that the predicate type of the first conjunct in a gapping construction is unified with predicates introduced by unary rules representing the elided verbs in the non-initial conjuncts. The predicate type reflects the argument structure of the clause, so the conjuncts with gapped verbs will have to realize the same type of arguments (for example a subject and an object) as the initial clause.

3.1 Incremental parsing and constituent structure

The constituent tree of the transitive sentence in (14) is assumed to be the flat structure in Figure 2a. The constituent structure is derived from the AVM of the parse tree, shown in Figure 2b. The step from parse tree to constituent tree involves the use of a feature STACK (Haugereid and Morey, 2012). In the following, the trees that will be presented, are parse trees, but they all have corresponding constituent trees.

(14) John-ga hon-o katta.
John-NOM book-ACC bought
John bought books.



(a) Constituent tree

(b) Parse tree

Figure 2: Illustration of constituent tree and parse tree of a transitive sentence in Japanese

The parse starts in the bottom left corner with a *START* symbol. This symbol has the features shown in (15). In the illustration, there are three valence features that all have negative values *arg1–*, *arg2–*, and *arg3–*. In addition, the value of the feature VBL is *synsem*, which means that it requires a verb.

(15)

$$\begin{bmatrix} \textit{START} & \\ \text{VAL} & \begin{bmatrix} \text{CMP1}\,|\,\text{LINK} & \textit{arg1–} \\ \text{CMP2}\,|\,\text{LINK} & \textit{arg2–} \\ \text{CMP3}\,|\,\text{LINK} & \textit{arg3–} \end{bmatrix} \\ \text{VBL} & \textit{synsem} \end{bmatrix}$$

The *START* symbol is combined with the subject *John-ga* and the direct object *hon-o*. There are separate valence rules for each of these functions, and they switch a negative link type to a positive. The rule for the direct object is shown in (16). It changes the negative link value *arg2–* in the first daughter to a positive link value in the mother *arg2+*.

(16)

$$\begin{bmatrix} \textit{cmp2-struc} & \\ \text{VAL} & \begin{bmatrix} \text{CMP1} & \boxed{1} \\ \text{CMP2} & \boxed{2}\begin{bmatrix}\text{LINK } \textit{arg2+}\end{bmatrix} \\ \text{CMP3} & \boxed{3} \end{bmatrix} \\ \text{VBL} & \boxed{4}\textit{synsem} \\ \text{ARGS} & \left\langle \begin{bmatrix} \text{VAL} & \begin{bmatrix} \text{CMP1} & \boxed{1} \\ \text{CMP2}\,|\,\text{LINK } \textit{arg2–} & \\ \text{CMP3} & \boxed{3} \end{bmatrix} \\ \text{VBL} & \boxed{4} \end{bmatrix}, \boxed{2} \right\rangle \end{bmatrix}$$

At the top of the tree in Figure 2b, the verb is realized. This is done by the verb rule shown in (17). The rule takes as its first daughter a structure that requires a verb, and as its second daughter a verb, and it produces a structure that has saturated the verb requirement (VBL *anti-synsem*). In addition, the rule unifies all the link types with the PRED type of the verb.

(17)

$$\begin{bmatrix} \textit{verb-struc} & \\ \text{VAL} & \boxed{1}\begin{bmatrix} \text{CMP1}\,|\,\text{LINK} & \boxed{2} \\ \text{CMP2}\,|\,\text{LINK} & \boxed{2} \\ \text{CMP3}\,|\,\text{LINK} & \boxed{2} \end{bmatrix} \\ \text{VBL} & \textit{anti-synsem} \\ \text{ARGS} & \left\langle \begin{bmatrix} \text{VAL} & \boxed{1} \\ \text{VBL} & \boxed{3} \end{bmatrix}, \boxed{3}\begin{bmatrix} \text{HEAD} & \textit{verb} \\ \text{LKEYS}\,|\,\text{KEYREL}\,|\,\text{PRED} & \boxed{2} \end{bmatrix} \right\rangle \end{bmatrix}$$

The lexical entry for the verb *ka* (‘buy’) is shown in (18). It only has an ORTH value, a HEAD value, and a PRED value. There are no VAL features or ARG-ST list.

(18) $$\begin{bmatrix} \textit{verb-lxm} & \\ \text{ORTH} & \left\langle \text{ka} \right\rangle \\ \text{HEAD} & \textit{verb} \\ \text{LKEYS}|\text{KEYREL}|\text{PRED} & \textit{buy\_prd} \end{bmatrix}$$

Instead of the regular valence requirements associated with verb lexical items, the verb is given a PRED value *buy_prd*, and it is the position of this type in a type hierarchy of subconstruction types, that determines which argument frames that are possible for the verb. A simplified type hierarchy involving the type *buy_prd* is shown in Figure 3.

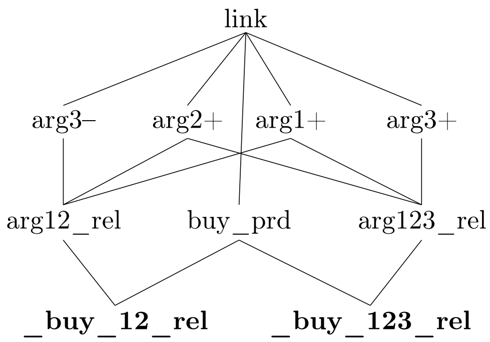


Figure 3: Type hierarchy of subconstruction types, argument frame types, and construction types

The hierarchy shows that the *buy_prd* type is compatible with two argument frames, a transitive frame *arg_12_rel*, and a ditransitive frame *arg_123_rel*. When the predicate is unified with one of these two frames, we get the construction types *_buy_12_rel* and *_buy_123_rel*, respectively. In this way, it is the type hierarchy of subconstruction types that determines which frames that are possible for a verb to enter.

The tree in Figure 4 shows how the linking types are changed from negative in the *START* node to positive in the top of the tree, and how the link types are unified with the PRED value of the verb. Since the types *arg1+*, *arg2+*, *arg3–*, and *buy_prd* are compatible (given the type hierarchy in Figure 3) the sentence is ultimately given a parse.

3.2 Analysis of gapping

SOV clause structure and backward gapping as demonstrated in (2)-(6) pose a challenge to the incremental left-to-right approach in Haugereid (2017). However, the

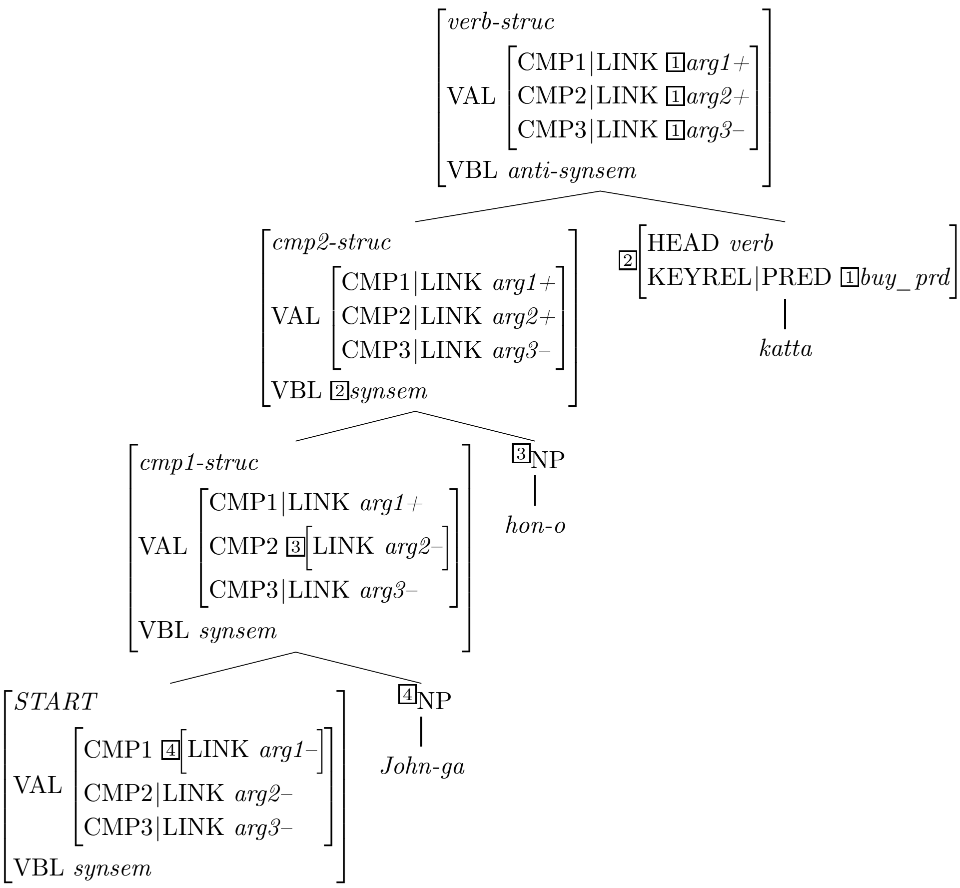


Figure 4: Parse tree for Japanese transitive sentence

constructional approach allows for the construction of an argument frame type that is underspecified with regard to the predicate of the main verb of the clause. This is shown in Figure 3, where the type *arg12_rel* is the result of the unification of three subconstruction types: *arg1*+, which is contributed by the rule that realizes the subject, *arg2*+, which is contributed by the rule that realizes the direct object, and *arg3*–, which shows that no indirect object has been realized. (If an indirect object is realized, the *arg3*– type will be replaced by *arg3*+, resulting in the argument frame type *arg123_rel*.) The tree in Figure 5 illustrates how the subconstruction types accumulate as the conjuncts in (2) are parsed.[4]

The parse starts in the bottom left corner with the structure *START* that has only negative subconstruction types (see (15)), represented in the tree in Figure 5 as an empty set. The rule that attaches the subject *John-ga* adds the subconstruction type *arg1*+, and the rule that attaches the object *hon-o* adds the type *arg2*+. When

[4]In order to make the representation compact, I have used sets to illustrate the accumulation of the subconstruction types in Figure 5. In reality, each subconstruction type is the value of a separate feature. The underlining of subconstruction types in the tree represents the unification of these types.

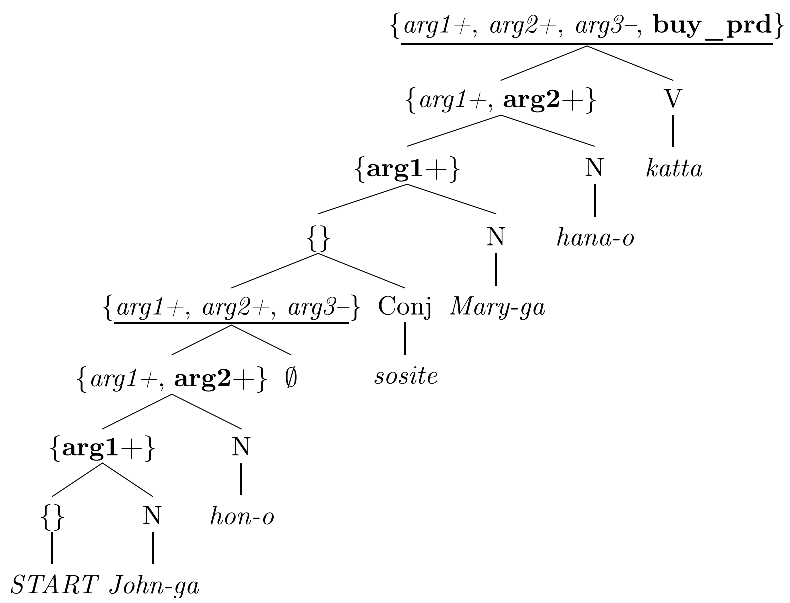


Figure 5: Accumulation of subconstruction types

no more arguments are attached, a unary gapping rule (see (19)) unifies the LINK values, here *arg1+*, *arg2+* and *arg3–*, with the PRED value of the KEYREL. (In the tree, this is marked by underlining the subconstruction types, and the elided verb is marked with the symbol $\emptyset$.) The KEYREL value is unified with the GAPREL value. The rule switches the VBL value from *synsem* in the daughter to *anti-synsem* in the mother. It also gets the HEAD feature GAPPING +.

(19)
$$\begin{bmatrix} \textit{verb-gapping-struc} & \\ \text{HEAD} & \begin{bmatrix}\text{GAPPING} +\end{bmatrix} \\ \text{VAL} & \boxed{1}\begin{bmatrix}\text{CMP1}\,|\,\text{LINK}\ \boxed{2} \\ \text{CMP2}\,|\,\text{LINK}\ \boxed{2} \\ \text{CMP3}\,|\,\text{LINK}\ \boxed{2}\end{bmatrix} \\ \text{VBL} & \textit{anti-synsem} \\ \text{LKEYS} & \begin{bmatrix}\text{KEYREL} & \boxed{3}\begin{bmatrix}\text{PRED}\ \boxed{2}\end{bmatrix} \\ \text{GAPREL} & \boxed{3}\end{bmatrix} \\ \text{ARGS} & \left\langle \begin{bmatrix}\text{VAL} & \boxed{1} \\ \text{VBL} & \textit{synsem}\end{bmatrix} \right\rangle \end{bmatrix}$$

At this point, the three subconstruction types are unified, resulting in the argument frame type *arg12_rel* (see the type hierarchy in Figure 3). The conjunct *sosite* initiates a new conjunct (see (20)), and it carries into the new clause the argument frame type from the gapping rule (see (21)).

(20)
$$\begin{bmatrix} \textit{conj-word} & \\ \text{ORTH} & \left\langle \textit{sosite} \right\rangle \\ \text{HEAD} & \begin{bmatrix} \textit{conj} & \\ \text{GAPPING} & + \end{bmatrix} \end{bmatrix}$$

(21)
$$\begin{bmatrix} \textit{coord-struc} & \\ \text{VAL} & \begin{bmatrix} \text{CMP1} \,|\, \text{LINK } \textit{arg1–} \\ \text{CMP2} \,|\, \text{LINK } \textit{arg2–} \\ \text{CMP3} \,|\, \text{LINK } \textit{arg3–} \end{bmatrix} \\ \text{VBL} & \textit{synsem} \\ \text{LKEYS} & \begin{bmatrix} \text{KEYREL} & \boxed{1} \end{bmatrix} \\ \text{ARGS} & \left\langle \begin{bmatrix} \text{HEAD} & \begin{bmatrix} \text{GAPPING} \boxed{2} \end{bmatrix} \\ \text{VBL} & \textit{anti-synsem} \\ \text{GAPREL} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \textit{coord-word} & \\ \text{HEAD} \,|\, \text{GAPPING} & \boxed{2} \end{bmatrix} \right\rangle \end{bmatrix}$$

The second clause is parsed in the same manner, and at the top of the tree, the rule that attaches the verb, unifies the predicate *buy_prd* with the subconstruction types of the second conjunct (*arg1*+, *arg2*+, *arg3*–), resulting in the predicate type *buy_12_rel*. (The unified subconstruction types are underlined at the top of the tree in Figure 5). The rule also unifies this predicate type with the argument frame type carried over from the first conjunct (*arg12_rel*). In this way, the identity of the two construction types is ensured, and the two clauses get the same predicate.

The MRS (Copestake *et al.*, 2005) for example (2), repeated below as (22) is given in Figure 6. The first *_buy_12_rel* predicate is the result of unifying the construction type of the first conjunct *arg_12_rel* with the construction type of the last conjunct *_buy_12_rel*.

(22)

John-ga	hon-o	sosite	Mary-ga	hana-o	katta.
John-NOM	book-ACC	and	Mary-NOM	flower-ACC	bought

'John bought books, and Mary flowers.'

The incremental subconstructional approach assumed in this paper is similar to the approach in Abeillé *et al.* (2014) in that the argument frame of the conjunct with the verb is unified with the argument frame of the conjuncts with the elided verbs. However, it differs from Abeillé *et al.* (2014), as well as other lexicalist approaches, in several respects. Firstly, this account is an incremental account. It differentiates between a parse tree (which is left branching, as shown in Figure

$$
\begin{bmatrix}
\textit{mrs} & \\
\text{TOP} & \boxed{0}\,h \\
\text{INDEX} & \boxed{1}\,e \\
\text{RELS} & \left\langle
\begin{array}{l}
\begin{bmatrix} \textit{named\_rel} & \\ \text{LBL} & \boxed{3}\,h \\ \text{CARG} & \textit{"John"} \\ \text{ARG0} & \boxed{5}\,x \end{bmatrix},
\begin{bmatrix} \textit{proper\_q\_rel} & \\ \text{LBL} & \boxed{6}\,h \\ \text{ARG0} & \boxed{5}\,x \\ \text{RSTR} & \boxed{7}\,h \\ \text{BODY} & \boxed{8}\,h \end{bmatrix},
\begin{bmatrix} \textit{\_buy\_12\_rel} & \\ \text{LBL} & \boxed{9}\,h \\ \text{ARG0} & \boxed{10}\,e \\ \text{ARG1} & \boxed{5}\,x \\ \text{ARG2} & \boxed{11}\,x \end{bmatrix},
\begin{bmatrix} \textit{\_book\_n\_rel} & \\ \text{LBL} & \boxed{12}\,h \\ \text{ARG0} & \boxed{11}\,x \end{bmatrix}, \\
\begin{bmatrix} \textit{indef\_q\_rel} & \\ \text{LBL} & \boxed{13}\,h \\ \text{ARG0} & \boxed{11}\,x \\ \text{RSTR} & \boxed{14}\,h \\ \text{BODY} & \boxed{15}\,h \end{bmatrix},
\begin{bmatrix} \textit{\_and\_c\_rel} & \\ \text{LBL} & \boxed{16}\,h \\ \text{C-ARG} & \boxed{17}\,i \\ \text{L-INDEX} & \boxed{10}\,e \\ \text{R-INDEX} & \boxed{1}\,e \end{bmatrix},
\begin{bmatrix} \textit{named\_rel} & \\ \text{LBL} & \boxed{18}\,h \\ \text{CARG} & \textit{"Mary"} \\ \text{ARG0} & \boxed{20}\,x \end{bmatrix},
\begin{bmatrix} \textit{proper\_q\_rel} & \\ \text{LBL} & \boxed{21}\,h \\ \text{ARG0} & \boxed{20}\,x \\ \text{RSTR} & \boxed{22}\,h \\ \text{BODY} & \boxed{23}\,h \end{bmatrix}, \\
\begin{bmatrix} \textit{\_buy\_12\_rel} & \\ \text{LBL} & \boxed{24}\,h \\ \text{ARG0} & \boxed{1}\,e \\ \text{ARG1} & \boxed{20}\,x \\ \text{ARG2} & \boxed{25}\,x \end{bmatrix},
\begin{bmatrix} \textit{\_flower\_n\_rel} & \\ \text{LBL} & \boxed{26}\,h \\ \text{ARG0} & \boxed{25}\,x \end{bmatrix},
\begin{bmatrix} \textit{indef\_q\_rel} & \\ \text{LBL} & \boxed{27}\,h \\ \text{ARG0} & \boxed{25}\,x \\ \text{RSTR} & \boxed{28}\,h \\ \text{BODY} & \boxed{29}\,h \end{bmatrix}
\end{array}
\right\rangle \\
\text{HCONS} & \left\langle
\begin{array}{l}
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{7}\,h \\ \text{LARG} & \boxed{3}\,h \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{14}\,h \\ \text{LARG} & \boxed{12}\,h \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{30}\,h \\ \text{LARG} & \boxed{9}\,h \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{22}\,h \\ \text{LARG} & \boxed{18}\,h \end{bmatrix}, \\
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{28}\,h \\ \text{LARG} & \boxed{26}\,h \end{bmatrix},
\begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{0}\,h \\ \text{LARG} & \boxed{24}\,h \end{bmatrix}
\end{array}
\right\rangle
\end{bmatrix}
$$

Figure 6: Semantic representation – Gapping

5) and a constituent tree, which is relatively standard (see Haugereid and Morey (2012)). Secondly, this approach assumes a hierarchy of subconstruction types (Haugereid, 2009), as illustrated in Figure 3. It is this hierarchy of subconstruction types that accounts for the argument frames in the grammar, not the constraints on the lexical items, and this makes it possible to parse a sentence without a verb, as illustrated in the first conjunct in Figure 5. In standard HPSG, including Abeillé *et al.* (2014), verbs are specified with an ARG-ST, and there is no generalization over ARG-ST lists that corresponds to the hierarchy of subconstruction types assumed in my approach. A third difference between the approach in this paper and Abeillé *et al.* (2014) is the semantics. In my approach, the construction type that results from the unification of the subconstruction types, becomes the predicate of the verb (and the elided verb). The semantics is in this way integrated with the syntax. In Abeillé *et al.* (2014) however, there are separate constraints accounting for the syntax and semantics of gapping.

4 Future work

The suggested method accounts for the data in (2)–(6). There will be some overgeneration with regard to adjuncts, since they are not reflected in the argument structure of the verb. One solution to that would be to let not only information about arguments, but also adjuncts be carried over to the next conjunct. This is a topic for further investigation. Another foreseeable problem with the approach is the fact that the verb does not appear in the first conjunct. This will increase the search space of the parser, although it will be constrained by the hierarchy of subconstruction types. The search space could be further restricted if the method were to be combined with some kind of statistical ”guesser” for each word that is added.

References

Abeillé, A., Bîlbîie, G., and Mouret, F. (2014). A romance perspective on gapping constructions. In H. C. Boas and F. Gonzálvez-García, editors, *Romance Perspectives on Construction Grammar*, pages 227–265. John Benjamins Publishing Company.

Beavers, J. and Sag, I. A. (2004). Coordinate ellipsis and apparent non-constituent coordination. In S. Müller, editor, *Proceedings of the HPSG-2004 Conference, Center for Computational Linguistics, Katholieke Universiteit Leuven*, pages 48–69. CSLI Publications, Stanford.

Chaves, R. P. (2005). A linearization-based approach to gapping. In G. Jäger, P. Monachesi, G. Penn, and S. Wintner, editors, *FG-MOL 2005: The 10th conference on Formal Grammar and The 9th Meeting on Mathematics of Language*, page 14. CSLI.

Copestake, A. (2002). *Implementing Typed Feature Structure Grammars*. CSLI publications.

Copestake, A., Flickinger, D., Pollard, C. J., and Sag, I. A. (2005). Minimal recursion semantics: an introduction. *Research on Language and Computation*, **3**(4), 281–332.

Crysmann, B. (2008). An asymmetric theory of peripheral sharing in HPSG: Conjunction reduction and coordination of unlikes. In G. Penn, editor, *Proceedings of FGVienna : The 8th Conference on Formal Grammar, Aug 16—17 2003, Vienna*, pages 45–64, Stanford. CSLI publications.

Haugereid, P. (2009). *Phrasal subconstructions: A constructionalist grammar design, exemplified with Norwegian and English*. Ph.D. thesis, Norwegian University of Science and Technology.

Haugereid, P. (2017). An incremental approach to gapping and conjunction reduction. In S. Müller, editor, *Proceedings of the 24th International Conference on Head-Driven Phrase Structure Grammar, University of Kentucky, Lexington*, pages 179–198, Stanford, CA. CSLI Publications.

Haugereid, P. and Morey, M. (2012). A left-branching grammar design for incremental parsing. In S. Müller, editor, *Proceedings of the 19th International Conference on Head-Driven Phrase Structure Grammar, Chungnam National University Daejeon*, pages 181–194.

Kathol, A. (1995). *Linearization-Based German Syntax*. Ph.D. thesis, Ohio State University.

Kato, K. (2006). *Japanese Gapping in Minimalist Syntax*. Ph.D. thesis, University of Washington.

Kim, Y.-J. and Cho, S.-Y. (2012). Tense and honorific interpretations in korean gapping construction: A constraint- and construction-based approach. In S. Müller, editor, *Proceedings of the 19th International Conference on Head-Driven Phrase Structure Grammar, Chungnam National University Daejeon*, pages 388–408.

Mouret, F. (2006). A phrase structure approach to argument cluster coordination. In S. Müller, editor, *The Proceedings of the 13th International Conference on Head-Driven Phrase Structure Grammar*, pages 247–267, Stanford. CSLI Publications.

Reape, M. (1994). Domain union and word order variation in German. In J. Nerbonne, K. Netter, and C. J. Pollard, editors, *German in Head-Driven Phrase Structure Grammar*, number 46 in CSLI Lecture Notes, pages 151–197. CSLI Publications, Stanford University.

Ross, J. R. (1970). Gapping and the order of constituents. In M. Bierwisch and K. Heidolph, editors, *Progress in Linguistics*, pages 249–259. The Hague: Mouton.

Sag, I. A., Gazdar, G., Wasow, T., and Weisler, S. (1985). Coordination and how to distinguish categories. *Natural Language & Linguistic Theory*, **3**(2), 117–171.

Yatabe, S. and Tanigawa, K. (2018). The fine structure of clausal right-node raising constructions in japanese. In S. Müller and F. Richter, editors, *Proceedings of the 25th International Conference on Head-Driven Phrase Structure Grammar, University of Tokyo*, pages 176–196, Stanford, CA. CSLI Publications.