Automatic Derivation of Semantic Representations for Thai Serial Verb Constructions: A Grammar-Based Approach

Vipasha Bansal
University of Washington
vipashab@uw.edu

Abstract

Deep semantic representations are useful for many NLU tasks (Droganova and Zeman, 2019; Schuster and Manning, 2016). Manual annotation to build these representations is time-consuming, and so automatic approaches are preferred (Droganova and Zeman, 2019; Bender et al., 2015). This paper demonstrates how rich semantic representations can be automatically derived for Thai Serial Verb Constructions (SVCs), where the semantic relationship between component verbs is not immediately clear from the surface forms. I present the first fully-implemented, unified analysis for Thai SVCs, deriving appropriate semantic representations (MRS; Copestake et al., 2005) from syntactic features, implemented within a DELPH-IN computational grammar (Slayden, 2009). This analysis increases verified coverage of SVCs by 73% and decreases ambiguity by 46%. The final grammar can be found at: https://github.com/VipashaB94/ThaiGrammar

1 Introduction

This paper presents the first fully-implemented analysis of a broad range of Thai SVCs in a computational grammar. An example of a Thai SVC is seen in (1).[1]

(1)	สุรี	ไป	เดิน	อ่าน	หนังสือ
	Suri	paj	dən	ʔà:n	nǎŋsɯ́ɯ
	Suri	go	walk	read	book

‘Suri went away from the speaker to read while walking’ (Thepkanjana 1986)

My grammar implementation uses HPSG (Pollard and Sag, 1994; Müller et al., 2021), and produces semantic representations in the Minimal Recursion Semantics (MRS) framework (Copestake et al., 2005). These representations model the semantic relationships in the construction, which are derived from the syntactic features of component verbs. The analysis was implemented on the basis of a DELPH-IN computational grammar, originally developed by Slayden (2009). The final grammar was tested against 216 development sentences, 205 regression sentences, and 85 held-out sentences, of which 77 were from naturally-occurring data. I show that this implementation increases verified coverage of Thai SVCs by 73% and decreases ambiguity by 46% on held-out data.

Semantic parsing is beneficial for performing various Natural Language Understanding (NLU) tasks such as biomedical text mining or open domain relation extraction (Schuster and Manning, 2016; Bender et al., 2015). Rich semantic representations can greatly improve the performance of systems on such tasks. For example, in dependency parsing, dependency trees containing deep semantic representations are more useful than surface-syntactic dependency trees (Droganova and Zeman, 2019; Schuster and Manning, 2016), which often rely too strongly on the surface structure of sentences, and do not show the relationships between content words (Schuster and Manning, 2016).

Arriving at these deep semantic representations requires complex semantic annotation (Droganova and Zeman, 2019), which can be either manual or grammar-driven. For example, the Enhanced (and Enhanced ++) Universal Dependency representations aim to make certain implicit relationships between content words more explicit by adding relations and augmenting relation names (Schuster and Manning, 2016). Alternatively, the English Resource Grammar (ERG; Flickinger 2000, 2011) takes a grammar-driven approach to produce compositional meaning annotations, and can successfully derive syntactic and semantic analyses for 85-95% of utterances in English text corpora (Bender et al., 2015).

[1]Most examples in this paper are drawn from previous work, but presented with slight modifications. In particular, the original proper name or pronoun has been changed to the name ‘Suri’, allowing for ease of implementation without materially changing the example. I also added Thai script and normalized the transcriptions and glosses.

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop), pages 318–333

Manual annotation, particularly for previously unannotated languages, is time-consuming and resource intensive, and so automatic approaches are extremely beneficial (Droganova and Zeman, 2019). Bender et al. (2015) argue that task- and domain-independent, automatically derivable methods to generate semantic representations would benefit the development of NLU systems, making them more comprehensive, consistent, and scalable. This can be achieved by using a compositional, linguistically-informed approach (Bender et al., 2015).

This paper focuses on the automatic derivation of semantic representations of a specific linguistic phenomenon which requires enrichment — Serial Verb Constructions (SVC). SVCs have been attested in numerous languages across West Africa, Central America, South-East Asia, and Oceania (Müller and Lipenkova, 2009). They can have a wide range of semantic interpretations, but the specific relationships between component verbs are not explicitly indicated by the surface forms; they are instead constrained by grammatical properties. By encoding these grammatical properties, we can get from the surface string to the semantic representation. Thai makes extensive use of SVCs – in Pongsutthi et al. (2013)'s study of 76 news articles (over 10,000 words) taken from the THAI-NEST corpus, 74.63% of the verb tokens were part of an SVC. Given their frequency, to successfully complete any NLU task for Thai, we must be able to deal with these constructions.

The analysis developed in this paper makes the implicit relationships between component verbs explicit, without the need for manual annotation. This implementation is the first step towards building an SVC library within the LinGO Grammar Matrix customization system (Bender et al., 2002, 2010; Zamaraeva et al., 2022), which will allow for efficient implementation of the phenomenon across typologically distinct languages.

2 Background

2.1 Definition of SVC

An SVC is any clause containing two or more verbs with no overt marker of coordination, subordination, or other type of syntactic dependency (Aikhenvald, 2006; Inman, 2019). They must be monoclausal, and each component verb must be able to appear as the only verb in the sentence and have the same tense, aspect, and polarity value (Aikhenvald, 2006; Haspelmath, 2016). They can encode a single event, subevents of a larger event, or two closely related events (*ibid*). Finally, they must be compositional — lexicalized or idiomatic forms, including verbal compounds, are not SVCs (Haspelmath, 2016; Pongsutthi et al., 2013).

2.2 Related Work

This paper builds on previous theoretical, but non-implemented, syntactic analyses of Thai SVCs. Sudmuk (2005) identifies eight types of Thai SVC and presents a unified LFG analysis encompassing each of these categories. Muansuwan (2002) uses HPSG to analyze a subset of Thai SVCs: Directional SVCs, Adjoining Constructions, and Aspectual Constructions. My analysis adapts Sudmuk's (2005) classification, incorporating data identified by Muansuwan (2002) and Thepkanjana (1986), as well as insights gained from fieldwork, to develop a comprehensive, categorization of Thai SVCs (Section 3) based on the semantic relationship between component verbs.[2]

Muansuwan (2002) presents independent analyses for each SVC subtype. In each case she uses valence-changing lexical rules; for Adjoining Constructions, she additionally proposes a type-hierarchy based on argument-sharing, and for Directional SVCs, she also uses a co-headed phrase structure rule, where a binary FIRST feature controls the ordering of component verbs. In an HPSG analysis of Mandarin Chinese SVCs (implemented within the TRALE system), Müller and Lipenkova (2009) present a series of non-headed phrase structure rules, incorporating aspectual properties to derive the semantics of each construction.

My analysis builds on each of these accounts, with a focus on creating a unified analysis which minimizes overgeneration and structural ambiguity. To achieve this, I incorporate information about the specific types and properties of component verbs in each SVC type (in addition to their argument-sharing properties). I also use examples from Thepkanjana (1986) and Diller (2006) for development data and additional context. This is the first computational implementation of Thai SVCs.

3 Data and Categorization

Based on both existing literature and my own consultations with a first language speaker of Thai,

[2]Constructions which violate the criteria in Section 2.1 are excluded.

I identify 5 semantic categories of Thai SVC, arranged in 3 argument-sharing configurations, creating a total of 7 SVC types (2-8). Sudmuk (2005) uses extraction and negation to demonstrate that these constructions are distinct from asyndetic coordination. Verbs in all Thai SVCs share at least one argument (*ibid*); in most cases, this is the subject (4a) and (5-9). In some resultative constructions, the object of the first verb is the subject of the second verb, referred to as switch-function SVCs (4b) (Aikhenvald, 2006). In sequential (2) and some purpose (3) SVCs, both the subject and the object are shared — both semantic interpretations are available in (3). Individual SVCs can also interact with additional verbs or SVCs to build longer, more complex structures, with more than one type of semantic relationship between component verbs. For example, (9) is a Deictic-Purpose SVC containing a Simultaneous SVC. This categorization forms the basis for the type-hierarchies and feature structures in Section 4.

4 Implementation and Analysis

4.1 Software

The analysis described in Section 4.3 was implemented on the basis of a computational precision grammar for Thai (Slayden, 2009), developed within the LinGO Grammar Matrix framework, situated within the DELPH-IN consortium. HPSG grammars in this framework are implemented as a collection of feature structures, lexical entries, and grammatical types, arranged into hierarchies which allow for inheritance (and multiple inheritance) of common features.

I used the LKB (Copestake, 2002) grammar development environment, which supports both parsing and generation, to build the feature structures, and `[incr tsdb()]` (Oepen and Flickinger, 1998) for regression testing.

Grammars implemented in this software environment include a set of files containing the lexicon, lexical types and features, and grammar rules required to parse and generate sentences. A total of 45 lexical types, 20 phrase-structure rule types, and 33 lexical rules were added to this grammar in order to model the SVCs and produce appropriate semantic representations. The final grammar implementation can be found at https://github.com/VipashaB94/ThaiGrammar.

(2) Sequential

สุรี	หา	ของขวัญ	พบ
Suri	hǎ:	khɔ̌:ŋ-khwǎn	phóp
Suri	seek	present	find

'Suri sought then found the present'
(Muansuwan 2002)

(3) Open-Purpose

สุรี	ผัด	ข้าว	กิน
Suri	phàt	khâ:w	kin
Suri	fry	rice	eat

a) 'Suri fried rice to eat (rice)'
b) 'Suri fried rice then ate (rice)'
(Sudmuk 2005)

(4) Resultative

a)

สุรี	กิน	ข้าว	อิ่ม
Suri	kin	khâ:w	ʔìm
Suri	eat	rice	be.full

'Suri ate rice therefore the rice was gone'

b)

สุรี	กิน	ข้าว	หมด
Suri	kin	khâ:w	mòd
Suri	eat	rice	be.gone

'Suri ate rice therefore the rice was gone'
(Muansuwan 2002)

(5) Deictic Purpose

สุรี	ไป	ซื้อ	หนังสือ
Suri	paj	sɯ́ɯ	nǎŋsɯ́ɯ
Suri	go	buy	book

'Suri went to buy a book'

(6) Simultaneous

รี	ยืน	เคาะ	ประตู
Suri	yɨ:n	khɔ́:ʔ	pràtu:
Suri	stand	knock	door

'Suri knocked on the door while standing'
(Sudmuk 2005)

(7) Direction-Deictic

สุรี	เดิน	ไป
Suri	dən	paj
Suri	walk	go

'Suri walked away from the speaker'
(Sudmuk 2005)

(8) Long Directional

สุรี	เดิน	ข้าม	สะพาน	กลับ	ไป
Suri	dən	khâam	saphaan	klàb	paj
Suri	walk	cross	bridge	return	go

'Suri walked, crossing the bridge, returning, away from the speaker' (Muansuwan 2002)

(9) Deictic-Purpose with Simultaneous SVC

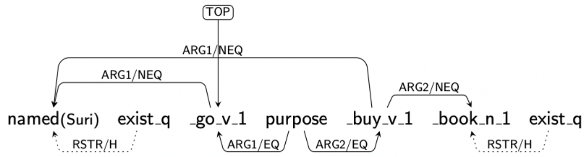
สุรี	ไป	เดิน	อ่าน	หนังสือ
Suri	paj	dən	ʔà:n	nǎŋsɯ́ɯ
Suri	go	walk	read	book

'Suri went away from the speaker to read while walking' (Thepkanjana 1986)

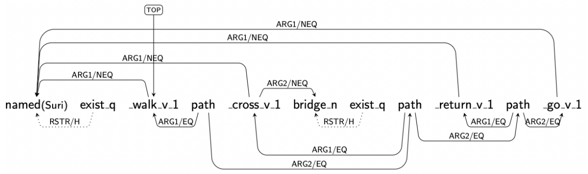
4.2 Target Semantic Representations

Fig.1 shows the target semantic representations for sentences (5), (8), and (9). The MRSes are formatted as dependency graphs (DMRS; Copestake, 2009), representing the links between each predicate and its arguments. For example, in Fig.1a, the verb *suu* ('to buy') has the predicate value *buy_v_1*. Its ARG1 (or subject) is the proper name *Suri*, and its ARG2 (or object) is the predicate *book_n_1*.

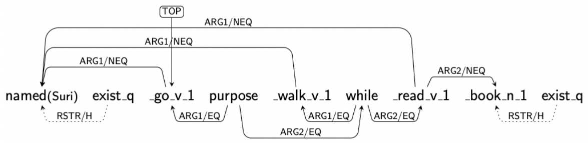
Sentence (5) is a Deictic-Purpose SVC contain-



(a) *MRS for Deictic-Purpose SVC in (5)*



(b) *MRS for Long-Directional SVC in (8)*



(c) *MRS for Interaction-Based SVC in (9)*

Figure 1: Target MRS Representations

ing two verbs, *paj* ('to go') and *suu* ('to buy'). Both verbs share a subject, and the second verb indicates the intention held while completing the action denoted by the first verb. Fig.1a shows these relationships: the verbs have PRED values *go_v_1* and *buy_v_1* respectively, and both take the proper name (PRED *named*) *Suri* as their initial argument. The predicate *purpose* then takes *paj* (V1) and *suu* (V2) as ARG1 and ARG2 respectively, making explicit the purposive relationship between them.

Sentence (8), represented by Fig.1b, is a longer SVC, consisting of four verbs, together indicating the direction in which the subject moves. Again, all verbs share a subject, each taking the predicate *named* as ARG1. There are three *path* predicates, which show pairwise relationships between verbs. The rightmost *path_rel* predicate takes the two final VPs *klab* (*return_v*) and *paj* (*go_v*) as arguments. The second takes *khâam* (*cross_v_1*) as ARG1 and the entire rightmost *path* predicate as ARG2, indicating that the direction of *khâam saphaan* ('cross the bridge') is the path created by *klab* and *paj* together. Similarly, the leftmost *path* predicate takes the initial verb *den* (*walk_v_1*) as ARG1 and the middle *path* predicate as ARG2.

Sentence (9) shows a Deictic-Purpose SVC where VP2 is a Simultaneous SVC. First, the predicate *while* takes *den* (*walk_v_1*) and *aan* (*read_v_1*) as arguments, showing that these actions occur at the same time. Then, the *purpose* predicate takes *go_v_1* and the entire *while* predicate as its arguments, indicating that the subject is 'going (away from the speaker)' with the intention to simultaneously walk and read. Again, all three verbs share the proper name subject *Suri*.

In this way, the semantic relationships between each verb in an SVC of any length can be explicitly modelled. In Section 4.3 I show how these representations can be systematically derived from the argument-sharing properties, individual features, and order of component verbs.

4.3 Derivation of Semantic Representations

In this analysis, I use a series of valence-changing lexical and phrase-structure rules which inherit from type-hierarchies based on syntactic and semantic properties of each SVC. Long Directional SVCs (8) and Direction-Deictic SVCs (7) both have directional semantics. However, Long Directionals are unique in that they can contain more than two verbs, and following Muansuwan (2002), have a recursive VP → VP VP structure. For all other categories (including Direction-Deictic SVCs), I follow Sudmuk (2005): SVCs have a complementation structure, where the initial verb is the head and selects for V2 (or VP2) (resulting in a VP → V V(VP) structure).

As Long Directional SVCs consist of two VPs, they require additional phrase-structure rules to combine with one another. For all other SVCs, lexical rules append an additional verbal complement to the initial verb's COMPS list, allowing them to combine using the existing Head-Complement Rule (Pollard and Sag, 1994). Interactions between SVCs can be analyzed using either lexical or phrase-structure rules based on their syntactic structure.

4.3.1 Features and Types

To control how various verbs, VPs, and SVCs interact with one another, a series of features are added to the HEAD value within the lexical entries for verbs. The Head Feature Principle (HFP) (Pollard and Sag, 1994) states that a mother node will have the same HEAD value as its head daughter. Therefore, these additional HEAD features are also inherited by VPs from their head verb (for Thai SVCs, this is always V1).

A binary SVC feature shows whether a verb forms an SVC after combining with its complements (or if a VP contains an SVC), while an SV-TYPE feature marks the semantic type of the SVC (e.g. *resultative*). An additional TYPE feature contains information about the specific properties of the verb itself. TYPE contains binary STATIVE and INTENTION features, as well as MDDP, which indicates if a verb is motion, directional, deictic, or posture (or none of these). For example, the verb *paj* ('to go') would have the TYPE value [MDDP *deictic*, STATIVE −, INTENTION +].

4.3.2 Lexical Rules

Lexical rules cross-inherit from type-hierarchies based on argument-sharing (Fig.2) and semantic (Fig.3) properties of each construction. This shows how SVCs with similar semantics but differing argument-sharing requirements (or vice versa) relate to one another and inherit their shared features.

Both hierarchies share *serial-verb-lex-rule* as their supertype. The next layer of the argument-sharing hierarchy differentiates based on the transitivity of V1 — this is when the additional verbal complement is added to the input verb's COMPS list. The remaining subtypes of rules across both hierarchies then add increasingly more specific constraints on how this added complement is integrated syntactically and semantically.

I illustrate by building the Deictic-Purpose SVC in (5) (and the DMRS in Fig.1a) using *deictic-purpose-lex-rule* (Fig.7), which inherits from *shared-subject-transitive-lex-rule*[3] (Fig.6) and *purpose-lex-rule* (Fig.5). These lexical rule-types (and their supertypes) are outlined below.[4]

In each rule-type, the input verb (represented by DTR), is V1 of the SVC. The topmost layer of the hierarchies, *serial-verb-lex-rule* (Appendix Fig.15) ensures the input verb has not been modified by an auxiliary verb or negative marker and is [SVC −], while the output verb is [SVC +]. Next, *transitive-v1-lex-rule* (Fig.4) places an additional verbal complement after a transitive input verb's existing NP complement. Both the subject and the existing complement of the input verb are identified with those of the resulting verb, ensuring that these remain unchanged. The added verbal complement is [OPT −], requiring it to be overt, and [AUX −, NEG −] to prevent auxiliary verbs and negative markers from intervening between the verbs.

Following Müller and Lipenkova (2009), the semantic relationship between the two verbs is introduced through the C-CONT feature, which has an item added to its RELS list.[5] This item takes each component verb as an argument, and has PRED value *purpose_rel* (which is introduced through *purpose-sem-lex-rule* (Fig.5)).

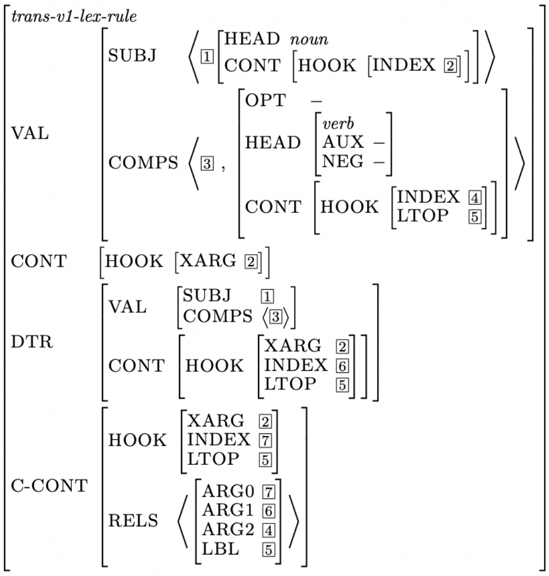


Figure 4: *transitive-v1-lex-rule*

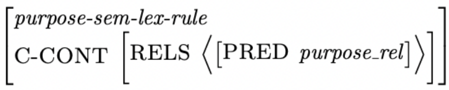


Figure 5: *purpose-sem-lex-rule*

Inheriting from *transitive-v1-lex-rule*, *shared-subject-trans-lex-rule* (Fig.6) identifies the XARG (external argument, a pointer to the subject) of the additional verbal complement with the XARG of the input verb and the INDEX of the subject NP, ensuring that the subject is shared by both verbs. It also ensures the added complement is a VP by specifying its COMPS list as empty.

Finally, *deictic-purpose-lex-rule* (Fig.7) constrains the individual properties of the component verbs. To form a Deictic-Purpose SVC, the input verb (V1) must be a deictic verb, while the complement VP is [SVC −], and cannot be deictic, stative, or without intent. The final construction is [SVTYPE *deictic-purpose*].

Returning to sentence (5), V1 is the deictic verb *paj* ('to go'), and therefore matches the requirements of the input verb and can undergo *deictic-purpose-lex-rule*. VP2 is headed by the verb *suu*

[3]Some Thai verbs, including *paj* ('to go') allow object dropping, and so (5) inherits from *shared-subject-transitive-lex-rule* even though we do not see an overt object for V1.

[4]Additional rule-types can be found in Appendix (A) and directly in the online grammar files.

[5]This is implemented in the grammar as a difference list.

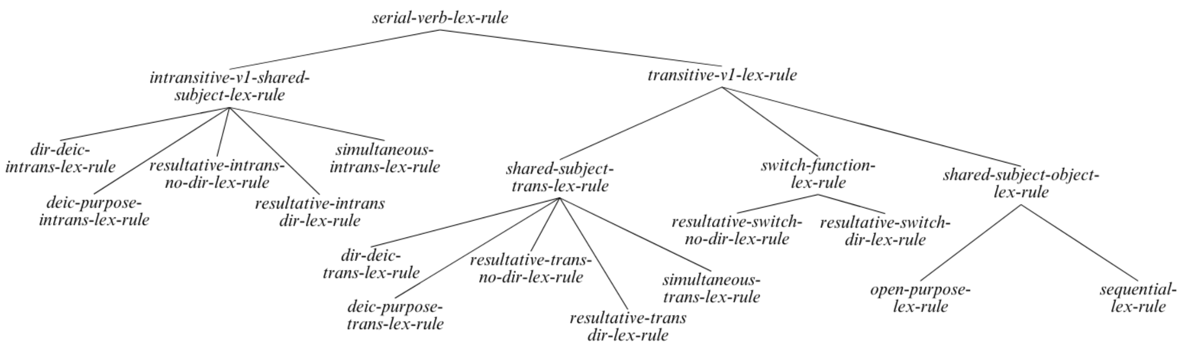


Figure 2: Argument-Sharing Type-Hierarchy

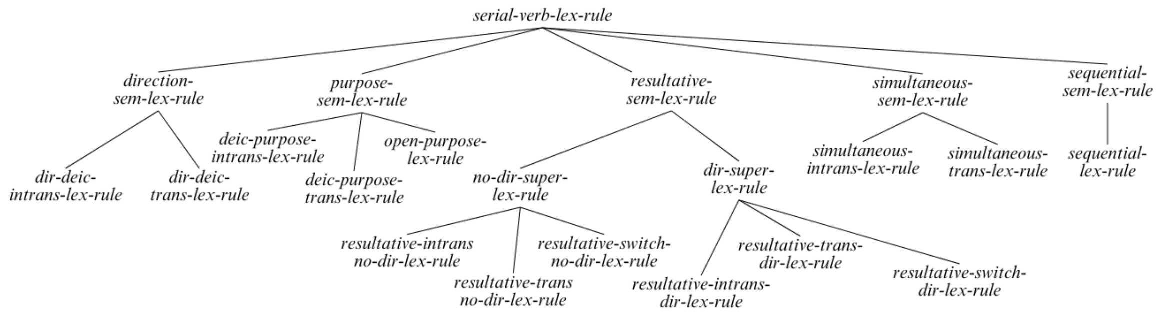


Figure 3: Semantic Type-Hierarchy

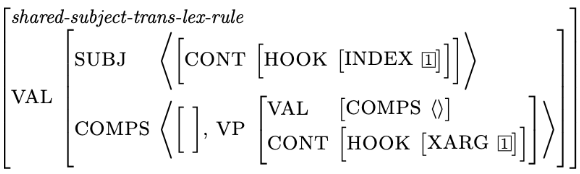


Figure 6: *shared-subject-transitive-lex-rule*

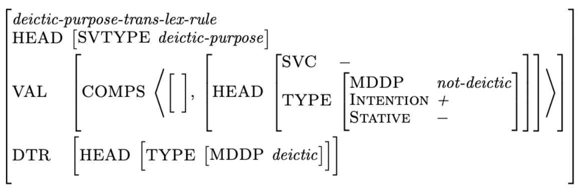


Figure 7: *deictic-purpose-transitive-lex-rule*

('to buy'), which matches each requirement placed on the complement VP, and is therefore added to V1's COMPS list via the lexical rule. Next, V1 *paj* can combine with VP2 *suu nangsuu* ('buy a book') through the Head-Complement Rule (Pollard and Sag, 1994), forming an overall VP. This VP takes the proper name *Suri* as its subject, which is shared by both verbs. This produces the DMRS in Fig.1a.

4.3.3 Phrase-Structure Rules

SVC-specific phrase-structure rules cross-inherit from type-hierarchies based on which (if any) component VPs contain an SVC themselves[6] (Fig.8), and on their semantics (Appendix Fig.30b).

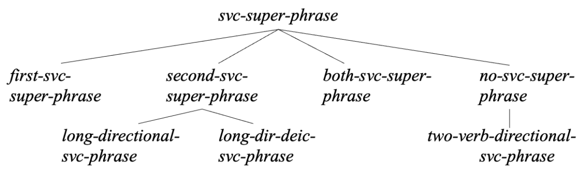


Figure 8: VP-Content Type-Hierarchy

Aside from some interaction-based SVCs, only Long Directional SVCs are formed using these additional phrase-structure rules, rather than lexical rules. They follow specific constraints: if a deictic verb is present, it must be the last verb in the SVC, if a motion verb is present, it must be the first verb in the SVC, and any directional verbs come in between (Muansuwan, 2002). A maximum of one motion and one deictic verb is permitted in the construction, though there can be multiple directional verbs. Additionally, if the SVC ends in a deictic

[6] All SVCs consisting of two VPs are shared-subject.

verb, the final VP is actually a Direction-Deictic SVC formed through *direction-deictic-svc-lex-rule* (Appendix A.4). The full syntactic structure is shown in Appendix B.4.

I illustrate by building the Long Directional SVC from (8) and deriving the MRS in Fig.1b. Since this SVC ends with a deictic verb, I use *long-direction-deictic-phrase* (Fig.12), which inherits from *second-svc-super-phrase* (Fig.10) and *motion-direction-sem-super-phrase* (Fig.11).

The topmost rule-type of the hierarchy, *svc-super-phrase* (Fig.9) takes two VP daughters, shown on its ARGS list. This rule-type identifies the XARGs of each component VP and the resulting VP with the INDEX of the subject NP, ensuring the component VPs share the same subject. Neither VP contains an auxiliary verb or negative marker. It also introduces the additional semantic relationship between them through the C-CONT feature — as with lexical rules, the additional item on the RELS list takes each component VP as an argument. Next, *second-svc-super-phrase* (Fig.10) marks VP1 as [SVC −] and VP2 as [SVC +].

The PRED value within the C-CONT, *path_rel*, is assigned by *direction-sem-super-phrase* (Fig.11). This rule-type also adds constraints that are common to all long-directional SVCs: VP1 must be headed by either a motion or a direction verb, while VP2 can only be headed by a direction verb (as motion verbs must be the initial verb in the construction, and therefore cannot appear in VP2). These constraints are reflected in the MDDP feature of of each component VP, which is inherited from the HEAD of the initial verb within each individual VP through the HFP (Pollard and Sag, 1994).

Finally, *long-direction-deictic-phrase* (Fig.12) specifices the SVTYPE of VP2 and of the overall construction, both of which are *direction-deictic*.

The construction in (8) is built from right to left. First, *klàb* (‘return’) and *paj* (‘go’) combine by lexical rule, forming a Direction-Deictic SVC headed by a directional verb. Next, the VP *khâam saphaan* (‘cross bridge’) uses *long-direction-deictic-phrase* to combine with this SVC, again forming a Direction-Deictic SVC headed by a directional verb. Then, *long-direction-deictic-phrase* is used one more time to combine *den* (‘walk’) with this SVC. As *den* is a motion verb, this VP is headed by a verb with [MDDP *motion*]. As *long-direction-deictic-phrase* requires VP2 to be headed by a directional verb, this rule cannot

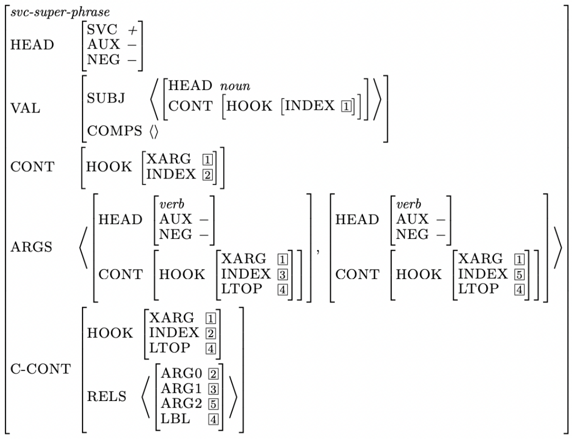


Figure 9: *Topmost Phrase-Structure Rule*

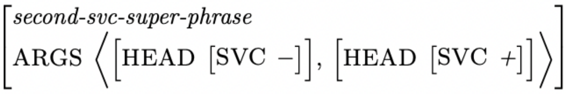


Figure 10: *second-svc-super-phrase*

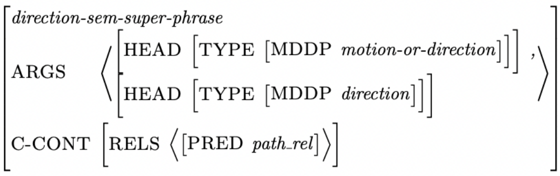


Figure 11: *direction-sem-super-phrase*

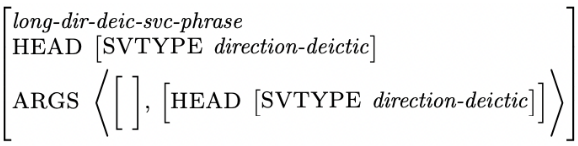


Figure 12: *long-direction-deictic-phrase*

be used to further combine the VP with any additional verbs, ensuring that the motion verb, when present, is first. The final VP then combines with the subject *Suri*, forming the DMRS in Fig.1c.[7]

4.3.4 Interactions

An SVC interaction is where one of the component VPs in an SVC is itself another SVC, producing a longer, more complex structure, often with more than one type of semantic relationship between component verbs. These interactions have strict constraints, and as before, the semantic relationships are compositionally derived from the

[7]Long Directional SVCs that do not end in a deictic verb can be built in a very similar manner using a combination of *short-directional-svc-phrase* and *long-directional-svc-phrase* (See Appendix B.4.1).

syntactic properties of the verbs. The same type-hierarchies presented in Sections 4.3.2 and 4.3.3 are used to account for the allowable combinations of SVCs while disallowing other combinations. Appendix A.1 and B.1 show these hierarchies with the additional interaction-based rules included.

For example, to build the Deictic-Purpose SVC with a Simultaneous SVC as VP2 in (9) and derive the DMRS in Fig.1c, I use *deictic-purpose-interact-trans-lex-rule* (Appendix Fig.37). This rule, like *deictic-purpose-lex-rule* (Fig.7), inherits from *shared-subject-transitive-lex-rule* (Fig.6) and *purpose-sem-lex-rule* (Fig.5), and specifies the input verb as [MDDP *deictic*]. The added VP complement however, has HEAD features [SVC +] and [SVTYPE *sim-dir-seq-pur*], allowing the deictic verb to select either a Simultaneous, Directional, Sequential, or Open-Purpose SVC (any of which must be constructed using another lexical rule before being selected by the deictic verb). In the case of (9), the deictic verb undergoes the lexical rule to have a Simultaneous SVC added to its COMPS list. This derives the DMRS in Fig.1c. Additional types of SVC interactions and associated grammar rules can be found directly in the implemented grammar.

5 Results

5.1 Regression Tests

I used Slayden's 2009 testsuite as regression tests to ensure that my additions to the grammar did not damage the analyses of other phenomena already implemented in the grammar. This testsuite contained 205 grammatical and ungrammatical sentences illustrating a variety of syntactic constructions. The results of these sentences when parsed with the final grammar were minimally different from the baseline, showing that existing functionality of the grammar was not significantly impacted.

5.2 Development Sentences

The development data was divided into four testsuites. The Main testsuite contains examples of each type of SVC that can be analyzed using lexical rules, and are not examples of SVC interactions. The Directionals testsuite contains examples of Long Directional SVCs, which require additional phrase-structure rules. The Interactions testsuite contains examples of SVCs combining with one another or additional verbs to create longer structures. Finally, the Coordination testsuite monitors non-asyndetic coordination, ensuring that sentences with overt coordination continue to parse despite the removal of asyndetic coordination from the grammar (and that they do not parse as SVCs). Table 1 shows the changes in verified coverage (number of grammatical examples that parse with accurate MRSes), overgeneration (number of ungrammatical examples that parse), and ambiguity (average number of parses per sentence) from baseline to the final grammar for each testsuite.

Testsuite	Verified Coverage		Over-Generation		Average Ambiguity	
	Baseline	Final	Baseline	Final	Baseline	Final
Main	3/82	77/82	32/56	9/56	1.51	1.81
Directional	0/15	13/15	4/7	1/7	3.82	1.08
Interactions	0/22	20/22	8/9	0/9	3.79	1.8
Coordination	17/24	24/24	1/1	0/1	2.38	1.63

Table 1: Results of Parsing Development Data

In the baseline run for the Main, Directional, and Interactions testsuites, almost every sentence, regardless of SVC category or grammaticality, parsed as asyndetic coordination - this led to low coverage and high overgeneration. For example, the Deictic Purpose SVC in (5) was originally assigned the the semantic representation in Fig.13. Here, a coordination predicate *and_c* takes *go_v_1* and *buy_v_1* as its left and right arguments respectively. Additionally, both verbs take *book_n_1* as their object, which does not make sense for *go_v_1*. However, the final grammar produces the DMRS shown in Fig.1a, which assigns the correct arguments to each component verb and indicates the purpose relationship between them.

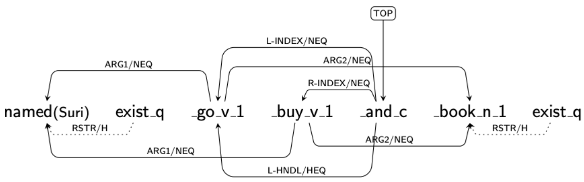


Figure 13: *Baseline, inaccurate DMRS for Sentence (5)*

Overall, the grammar improves on the baseline by giving the SVC examples correct semantic representations and reduces ambiguity in directionals and interactions by 71% and 53% respectively. While ambiguity for the Main testsuite increased slightly, there are significantly more verified parses and less overgeneration. Coordination behaves as expected.

Some ambiguity in the results is to be expected. First, some SVCs can legitimately have more than one interpretation, as seen in sentence (3). Second, some restrictions on interpretation are entirely pragmatic — for example, all Resultative SVCs with a transitive V1 can syntactically be either a shared-subject SVC or a switch-function SVC and will parse as such; the determination of which interpretation is grammatical is pragmatic,[8] and could be addressed in future through Redwoods-style treebanking (Toutanova et al., 2005) to support parse selection.

5.3 Held-Out Sentences

I gathered 71 held-out sentences from naturally-occurring data found in publicly available short stories and online language learning material (not related specifically to SVCs). An additional 14 held-out sentences were sourced from Pongsutthi et al. (2013) and Takahashi (2009), which were not consulted until after developing and implementing this analysis. Since Pongsutthi et al. (2013) is a corpus study based on Thai news articles, the 6 sentences taken from this paper are also considered naturally occurring.

The naturally-occurring sentences are often very complex, containing syntactic phenomena, such as topicalization, that are not currently implemented within the grammar and are beyond the scope of this project. Therefore, in order to avoid sentences failing due to unrelated causes, which would not allow for accurate testing of the grammar functionality with regards to the SVC implementation described here, the sentences were simplified to contain just relevant verbs and arguments. The substance of the SVC was not altered. Table 2. shows the results of parsing the held-out testsuite. We again see a significant improvement from the baseline grammar, with a 73% increase in verified coverage and a 46% decrease in ambiguity.

Verified Coverage		Average Ambiguity	
Baseline	Final	Baseline	Final
1/85	63/85	5.22	2.81

Table 2: Results of Parsing Held-Out Data

[8]Although pragmatics play an important role in SVC acceptability, Thai is more constrained than English in terms of forcing verbs into atypical readings based on context alone, particularly in SVCs. Therefore, it is unlikely that coverage is lost through under-generalization of verb types. For example, *lôm* ('fall'), which is [INTENTION −], cannot act as V2 of a purpose SVC, even in a specific situation where the subject falls intentionally - this would need to be expressed overtly.

6 Conclusion

This paper has demonstrated how deep semantic representations of Thai SVCs can be automatically derived from syntactic properties of component verbs and the structure of the phrase as a whole. This was implemented into a computational grammar using an HPSG analysis, and tested against development and held-out sentences. I showed that this analysis can successfully account for Thai SVCs, increasing accuracy and reducing overgeneration and spurious ambiguity in both development and held-out data. This allows for the creation of richer, more precise semantic representations of Thai SVCs, which explicitly model the relationship between component verbs.

The LinGO Grammar Matrix (Bender et al., 2002, 2010; Zamaraeva et al., 2022) both draws on and supports typological work (Bender, 2016). Its goal is to combine typological research and syntactic analysis, allowing for both cross-linguistic generalizations and language-specific constraints, in order to map from surface strings to semantic representations (Bender, 2016). This analysis follows this approach, allowing for flexibility in argument-sharing, constituent structure, and verbal features used for derivation, while situated within the typological constraints presented in Section 2.1.

7 Limitations

The main limitation of this analysis is that it models the prestige variety of Thai, and does not account for dialectal differences amongst speakers. Therefore, some speakers may have different grammaticality judgements on which SVCs can be used than what has been presented here. Additionally, SVCs involving ditransitive verbs were not included in this analysis. However, results from the held-out data show that in its current form, the analysis can handle the appearance of ditransitive verbs as V2. Extending the analysis to allow for ditransitive verbs as V1 is left to future work.

8 Acknowledgments

I would like to thank Emily Bender, for her extensive guidance and support on this project, and Yaovarat Rutheerayuth, for her valuable help with and insight into the Thai language. I am also grateful to the various researchers who have provided feedback and advice during grammar development, and to the anonymous reviewers for their helpful comments.

References

Alexandra Y. Aikhenvald. 2006. Serial verb constructions in typological perspective. In Alexandra Y. Aikhenvald and R.M.W. Dixon, editors, *Serial Verb Constructions: A Crosslinguistic Typology*. Oxford University Press.

Emily Bender, Scott Drellishak, Antske Fokkens, Laurie Poulson, and Safiyyah Saleem. 2010. Grammar customization. *Research on Language and Computation*, 8:23–72.

Emily M. Bender. 2008. Evaluating a crosslinguistic grammar resource: A case study of wambaya. In *Proceedings of ACL-08: HLT*, pages 977–985, Columbus, Ohio. Association for Computational Linguistics.

Emily M. Bender. 2016. Linguistic typology in natural language processing. *Linguistic Typology*, 20(3):645–660.

Emily M. Bender, Dan Flickinger, and Stephan Oepen. 2002. The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In *COLING-02: Grammar Engineering and Evaluation*.

Emily M. Bender, Dan Flickinger, Stephan Oepen, Woodley Packard, and Ann Copestake. 2015. Layers of interpretation: On grammar and compositionality. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 239–249, London, UK. Association for Computational Linguistics.

Ann Copestake. 2002. *Implementing Typed Feature Structure Grammars*. CSLI Publications, Stanford, CA.

Ann Copestake. 2009. **Invited Talk:** slacker semantics: Why superficiality, dependency and avoidance of commitment can be the right way to go. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 1–9, Athens, Greece. Association for Computational Linguistics.

Ann Copestake, Dan Flickinger, Ivan A. Sag, and Carl Pollard. 2005. Minimal recursion semantics: An introduction. *Research on Language and Computation*, 3.

A.V.N. Diller. 2006. Thai serial verbs: Cohesion and culture. In Alexandra Y. Aikhenvald and R.M.W. Dixon, editors, *Serial Verb Constructions: A Crosslinguistic Typology*. Oxford University Press.

Kira Droganova and Daniel Zeman. 2019. Towards deep Universal Dependencies. In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*, pages 144–152, Paris, France. Association for Computational Linguistics.

Martin Haspelmath. 2016. The serial verb construction: Comparative concept and cross-linguistic generalizations. *Language and Linguistics*, 17:291–319.

David. Inman. 2019. *Multi-predicate Constructions in Nuuchahnulth*. Ph.D. thesis, The University of Washington.

Nuttanart Muansuwan. 2002. *Verb Complexes in Thai*. Ph.D. thesis, University at Buffalo, The State University of New York.

Stefan Müller, Anne Abeille, Robert D. Borsley, and Jean-Pierre Koenig. 2021. *Head-Driven Phrase Structure Grammar: The handbook*. Empirically Oriented Theoretical Morphology and Syntax. Language Science Press.

Stefan Müller and Janna Lipenkova. 2009. Serial verb constructions in Chinese: An HPSG account. In *Proceedings of the 16th International Conference on Head-Driven Phrase Structure Grammar, University of Göttingen, Germany*, pages 234–254, Stanford. CSLI Publications.

Stephan Oepen and Dan Flickinger. 1998. Towards systematic grammar profiling.test suite technology 10 years after. *Journal of Computer Speech and Language*, 12:411–435.

Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press.

Mancharee Pongsutthi, Nongnuch Ketui, and Chutamanee Onsuwan. 2013. Thai serial verb constructions: A corpus-based study. In *10th International Symposium on Natural Language Processing (SNLP)*.

Sebastian Schuster and Christopher D. Manning. 2016. Enhanced English Universal Dependencies: An improved representation for natural language understanding tasks. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2371–2378, Portorož, Slovenia. European Language Resources Association (ELRA).

Glenn Slayden. 2009. Thai resource grammar.

Cholthicha. Sudmuk. 2005. *The Syntax and Semantics of Serial Verb Constructions in Thai*. Ph.D. thesis, The University of Texas at Austin.

Kiyoko Takahashi. 2009. Basic serial verb constructions in thai. *Journal of the Southeast Asian Linguistics Society*, 1:215–229.

Kingkarn Thepkanjana. 1986. *Serial Verb Constructions in Thai*. Ph.D. thesis, University of Michigan.

Kristina Toutanova, Christopher Manning, Dan Flickinger, and Stephan Oepen. 2005. Stochastic hpsg parse disambiguation using the redwoods corpus. *Research on Language and Computation*, 3:83–105.

Olga Zamaraeva, Chris Curtis, Guy Emerson, Antske Fokkens, Michael Goodman, Kristen Howell, T.J. Trimble, and Emily M. Bender. 2022. 20 years of the grammar matrix: cross-linguistic hypothesis testing of increasingly complex interactions. *Journal of Language Modelling*, 10(1):49–137.

A Further Information on Lexical Rule Type-Hierarchies and Feature Structures

A.1 Lexical Rule Type-Hierarchies

Fig.14 shows the full argument-sharing and semantic type-hierarchies and inheritance structures for lexical rules, including the interaction-based lexical rules (shown in bold) which were not included in the hierarchies in Section 4.3.2.

A.2 Argument-Sharing Lexical Rule-Types

This section illustrates the feature-structures for the non-leaf lexical rule-types in the Argument-Sharing hierarchy in Fig.14a. The topmost layer of the hierarchy, *serial-verb-lex-rule* (Fig.15), ensures that neither the input verb nor the output verb is an auxiliary verb or negative marker and that the input verb has not already been modified to form an SVC (while the output verb has).

The next layer is dependent on the transitivity of the input verb. Both *intransitive-v1-shared-subject-lex-rule* (Fig.16) and *transitive-v1-lex-rule* (Fig.17) add a verbal complement to the end of its COMPS list. The subject of the input verb is identified with that of the output verb, ensuring that it remains unchanged. The semantic relationship between the two verbs is introduced through the C-CONT feature, which has an item added to its RELS list. This item takes each component verb as an argument.

The next layer constrains the argument-sharing properties of the two verbs[9] by identifying the relevant valence features of the input verb with those of its verbal complement (Fig18-20).

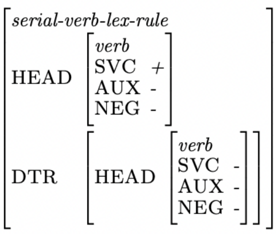


Figure 15: *serial-verb-lex-rule*

[9]If V1 is intransitive, both verbs must share the same subject. Therefore *intransitive-v1-shared-subject-lex-rule* (Fig.16) is a single rule-type that both adds the verbal complement and defines its argument-sharing constraints.

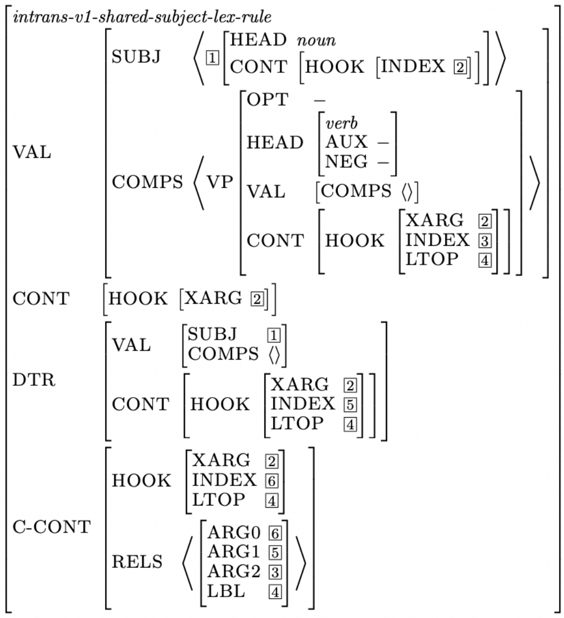


Figure 16: *intransitive-v1-shared-subject-lex-rule*

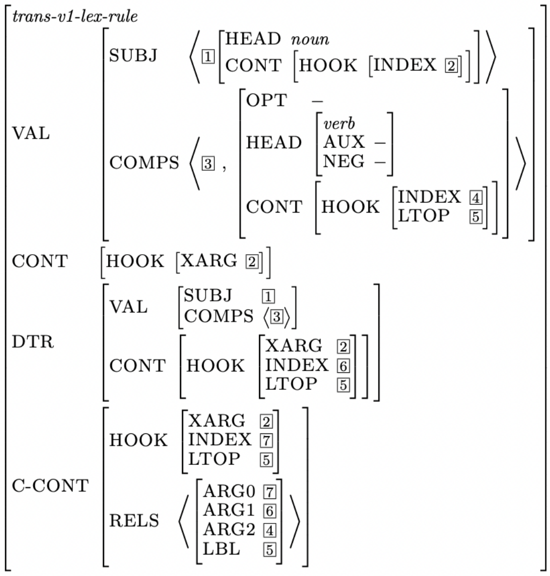


Figure 17: *transitive-v1-lex-rule*

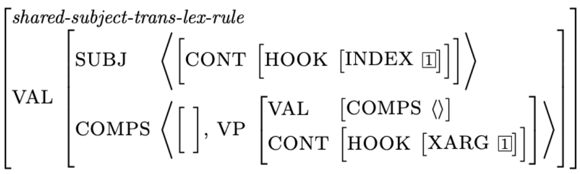
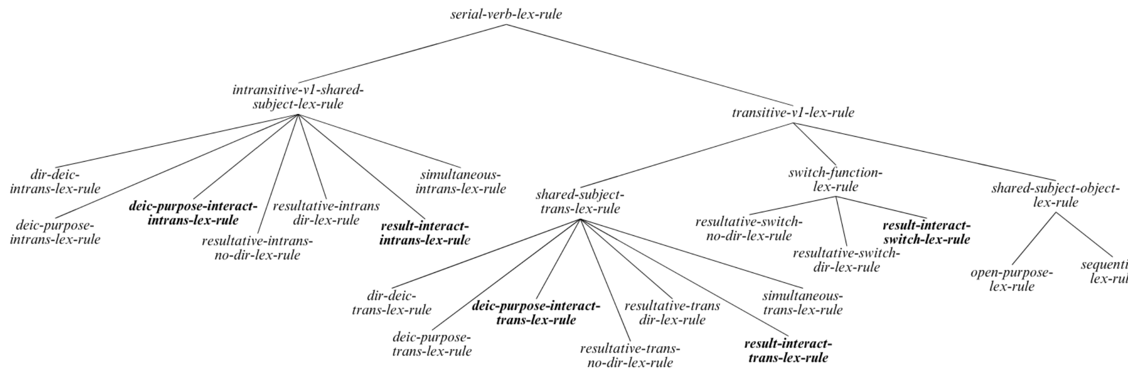
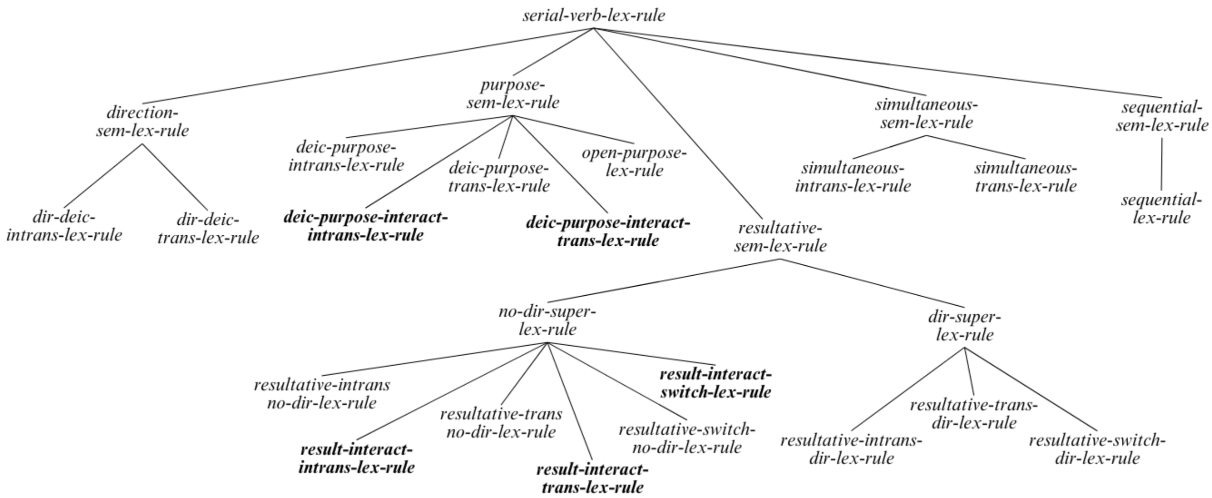


Figure 18: *shared-subject-transitive-lex-rule*



(a) Full Argument-Sharing Lexical Rule Type-Hierarchy



(b) Full Semantic Lexical Rule Type-Hierarchy

Figure 14: Full Argument-Sharing and Semantic Type-Hierarchies for Lexical Rules

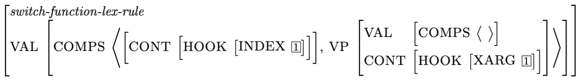


Figure 19: *switch-function-lex-rule*

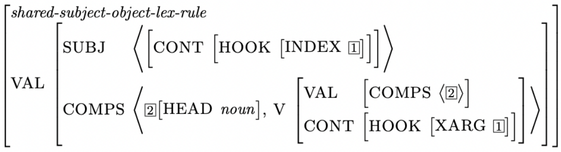


Figure 20: *shared-subject-object-lex-rule*

A.3 Semantic Lexical Rule-Types

This section illustrates the feature-structures for the non-leaf lexical rule-types in the semantic type-hierarchy for lexical rules in Fig.14b. In each case, they indicate the semantic relationship between the two verbs through the PRED value of the item added to the RELS list of the C-CONT. Resultative SVCs have an extra layer in the hierarchy, as the constraints on V2 differ based on whether the initial verb is a motion/direction verb or not. These specific constraints can be found directly in the implemented grammar.

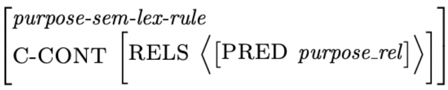


Figure 21: *purpose-sem-lex-rule*

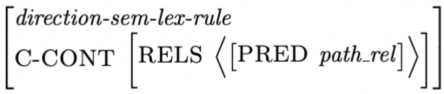


Figure 22: *direction-sem-lex-rule*

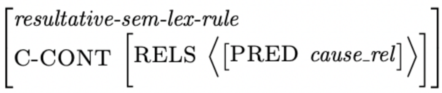


Figure 23: *resultative-sem-lex-rule*

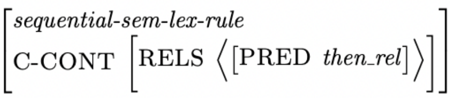


Figure 24: *sequential-sem-lex-rule*

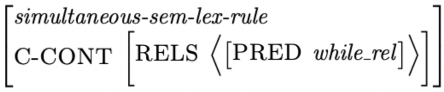


Figure 25: *simultaneous-sem-lex-rule*

A.4 Deriving a Direction-Deictic SVC by Lexical Rule

Section 4.3.2 demonstrated the derivation of a Deictic Purpose SVC by using *deictic-purpose-transitive-lex-rule*. To provide a further example, here we will derive a Direction-Deictic SVC, such as those in examples (10) and (11):

(10)	รี	ขี่	ม้า	ไป
Suri	khì:	mâ:	paj
Suri	ride	horse	go

‘Suri rode a horse away from the speaker’

(11)	สุรี	ข้าม	สะพาน	ไป
Suri	khâam	saphaan	paj
Suri	cross	bridge	go

‘Suri crossed the bridge away from the speaker’

In a Direction-Deictic SVC, V1 can be either a motion (10) or direction (11) verb, while V2 must be a deictic verb. Both sentences can be derived using *dir-deic-trans-lex-rule*, which cross-inherits from *shared-subject-trans-lex-rule* (Fig.18) and *direction-sem-lex-rule* (Fig.22), shown in Fig.26.

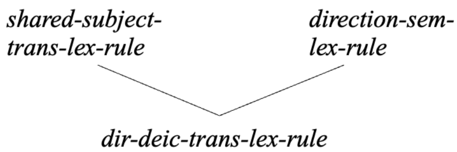


Figure 26: Cross-Inheritance for *dir-deic-trans-lex-rule*

In this way, the argument-sharing properties and the semantic relationship between the two verbs are inherited. Therefore, *dir-deic-trans-lex-rule* (Fig.27) is responsible only for constraining the specific properties of each verb (as are all other leaf nodes in the hierarchies above). The input verb (represented by DTR) has TYPE value [MDDP *motion-or-direction*], while the added VP complement is [MDDP *deictic*].

$$\begin{bmatrix} \textit{dir-deic-trans-lex-rule} \\ \text{HEAD} \ [\text{SVTYPE} \ \textit{direction-deictic}] \\ \text{VAL} \ \left[\text{COMPS} \left\langle [\,], \left[\text{HEAD} \begin{bmatrix}\text{SVC} & - \\ \text{TYPE} & [\text{MDDP} \ \textit{deictic}]\end{bmatrix}\right]\right\rangle\right] \\ \text{DTR} \ \left[\text{HEAD} \ [\text{TYPE} \ [\text{MDDP} \ \textit{motion-or-direction}]]\right] \end{bmatrix}$$

Figure 27: *dir-deic-trans-lex-rule*

This rule therefore allows both *khì:* (‘ride’) and *khâam* (‘cross’) to act as input verbs (with MDDP values of *motion* and *direction* respectively). The deictic verb *paj* (‘go’) can then be added to the input verb’s COMPS list, after the existing complement (*mâ:* (‘horse’) or *saphaan* (‘bridge’)). Due to the Head Feature Principle, the final VP for each example will have the following HEAD values:

$$\begin{bmatrix} \textit{khì: mâ: paj} \\ \text{HEAD} \begin{bmatrix} \text{SVC} & + \\ \text{AUX} & - \\ \text{NEG} & - \\ \text{SVTYPE} & \textit{direction-deictic} \\ \text{TYPE} & \begin{bmatrix}\text{MDDP} & \textit{motion} \\ \text{INTENTION} & + \\ \text{STATIVE} & -\end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Figure 28: HEAD value for (10)

$$\begin{bmatrix} \textit{khâam saphaan paj} \\ \text{HEAD} \begin{bmatrix} \text{SVC} & + \\ \text{AUX} & - \\ \text{NEG} & - \\ \text{SVTYPE} & \textit{direction-deictic} \\ \text{TYPE} & \begin{bmatrix}\text{MDDP} & \textit{direction} \\ \text{INTENTION} & + \\ \text{STATIVE} & -\end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Figure 29: HEAD value for (11)

The final VP combines with the subject *Suri*, which is shared by both verbs due to the constraints inherited from *shared-subject-trans-lex-rule*.

Lexical rules for other SVC types (such as Resultative or Sequential SVCs), and their associated constraints, can be found in the thai.tdl file in the implemented grammar.

B Further Information on Phrase-Structure Rule Type-Hierarchies and Feature Structures

B.1 Phrase-Structure Rule Type-Hierarchies

Fig.30 shows the full VP-content type-hierarchy for phrase-structure rules, including the interaction based rules (shown in bold) which were not included in the hierarchy in Section 4.3.3. It also includes the full semantic type-hierarchy for phrase-structure rules. With the exception of Directional SVCs, these phrase-structure rules are used mainly to allow existing SVCs to combine with each other or with additional verbs.

B.2 VP-Content Phrase-Structure Rule-Types

This section shows the feature-structures for the phrase-structure rule-types in the VP-content hierarchy in Fig.30a. The topmost layer of the hierarchy, *svc-super-phrase* (Fig.9) was shown in Section 4.3.3, and takes two VP daughters, shown on its ARGS list.

The rule-types in the next layer of the VP-Content hierarchy (Fig.31) define which (if any) of the component VPs contain an SVC. This is based on the VP’s binary [SVC] feature, inherited from the head verb of the VP through the Head Feature Principle (Pollard and Sag, 1994). The remaining layers of rule-types constrain the specific properties of component VPs for each SVC type.

$$\begin{bmatrix} \textit{first-svc-super-phrase} \\ \text{ARGS} \left\langle [\text{HEAD} \ [\text{SVC} \ +]], [\text{HEAD} \ [\text{SVC} \ -]] \right\rangle \end{bmatrix}$$

(a) *first-svc-super-phrase*

$$\begin{bmatrix} \textit{second-svc-super-phrase} \\ \text{ARGS} \left\langle [\text{HEAD} \ [\text{SVC} \ -]], [\text{HEAD} \ [\text{SVC} \ +]] \right\rangle \end{bmatrix}$$

(b) *second-svc-super-phrase*

$$\begin{bmatrix} \textit{both-svc-super-phrase} \\ \text{ARGS} \left\langle [\text{HEAD} \ [\text{SVC} \ +]], [\text{HEAD} \ [\text{SVC} \ +]] \right\rangle \end{bmatrix}$$

(c) *both-svc-super-phrase*

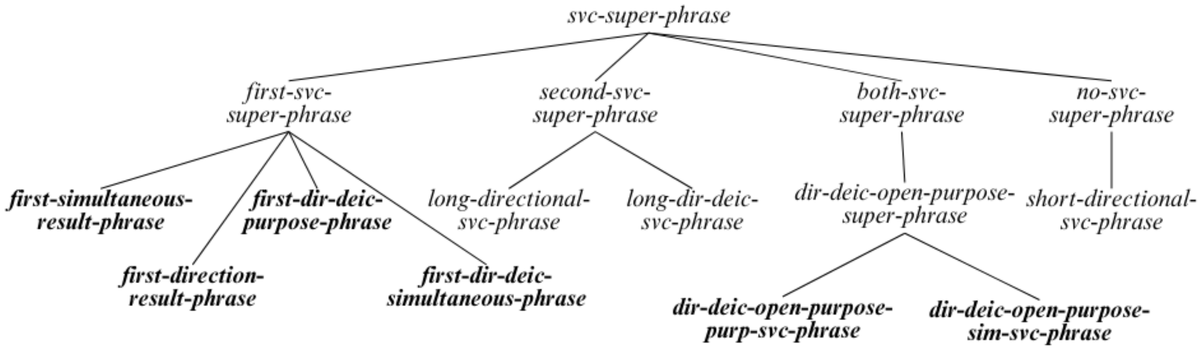
$$\begin{bmatrix} \textit{no-svc-super-phrase} \\ \text{ARGS} \left\langle [\text{HEAD} \ [\text{SVC} \ -]], [\text{HEAD} \ [\text{SVC} \ -]] \right\rangle \end{bmatrix}$$

(d) *no-svc-super-phrase*

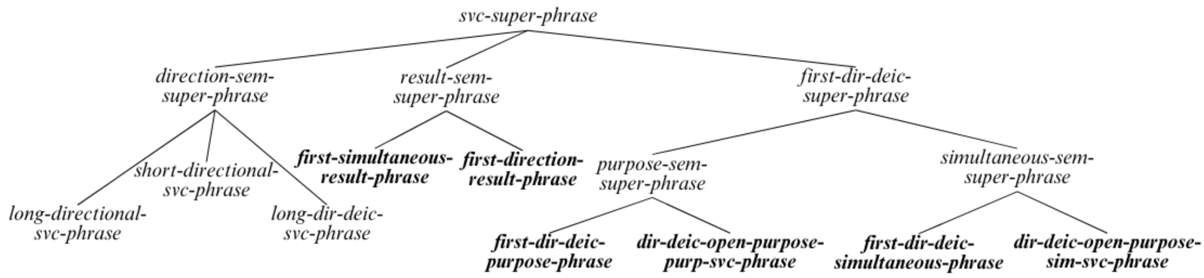
Figure 31: Rule-Types Defining SVC Content of VPs

B.3 Semantic Phrase-Structure Rule-Types

Each rule-type in this hierarchy indicates the semantic relationship between the two verbs through the PRED value of the item added to the RELS list of the C-CONT. *Direction-sem-super-phrase* (Fig.11) was shown in Section 4.3.3, and due to their close similarity with semantic lexical rule-types, the others have not been shown here, but can be found



(a) Full VP Content Phrase-Structure Rule Type-Hierarchy



(b) Full Semantic Phrase-Structure Rule Type-Hierarchy

Figure 30: Full VP Content and Semantic Type-Hierarchies for Phrase-Structure Rules

directly in the implemented grammar.

B.4 Long Directional SVCs

In Section 4.3.3 I argued that Long Directional SVCs have a recursive VP → VP VP structure, but that when the SVC ends in a deictic verb, the last pair of verbs actually forms a Direction-Deictic SVC, with a VP → V VP structure.

This is based on Muansuwan's (2002) adverb placement test to identify VP boundaries. She argues that adverbs can only appear at the end of a VP. In Directional SVCs, adverbs can intervene between each verb, except preceding a deictic verb (Muansuwan, 2002). Following this, sentence (8) (and other Long Directional SVCs) have the syntactic structure in Fig.32.

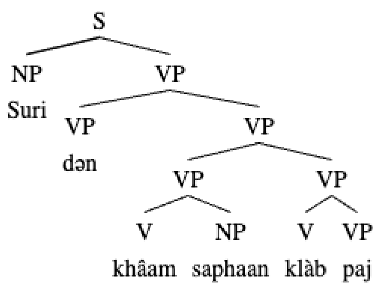


Figure 32: Structure for Long Directional SVC in (8)

B.4.1 Further Examples of Deriving Long Directional SVCs

Section 4.3.3 showed how *dir-deic-trans-lex-rule* (Fig.27) and *long-dir-deic-svc-phrase* (Fig.12) are used together to build a Long Directional SVC ending in a deictic verb. This section shows how phrase structure rules can be used to build Long-Directional SVCs which do not end in a deictic verb, such as sentence (12).

(12) สุรี เดิน ข้าม สะพาน กลับ บ้าน
Suri dən khâam saphaan klàb baan
Suri walk cross bridge return home
'Suri walked, crossing the bridge, returning home'

The presence of the final deictic verb affects which SVCs the final construction can interact with. Therefore, Long Directional SVCs without a deictic verb are [SVTYPE *directional*] (rather than [SVTYPE *direction-deictic*]). They are analyzed using a combination of *short-directional-svc-phrase* and *long-directional-svc-phrase*.

These two rules work together to build a Directional SVC in the same way as those in Section 4.3.3: *short-directional-svc-phrase* (Fig.34) will be used to combine the two rightmost verbs (neither of which contain an SVC), and then *long-directional-*

svc-phrase (Fig.36) is recursively applied to add verbs to the resulting VP, until there is a motion verb. The cross-inheritance and feature structures for each rule are shown in Figs.33-36 below.

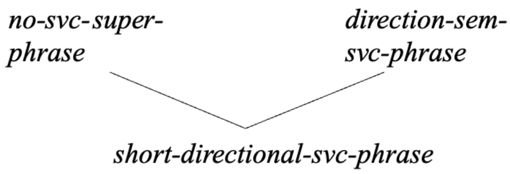


Figure 33: Cross-Inheritance for *short-directional-svc-phrase*

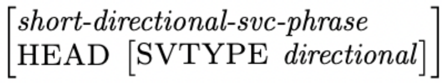


Figure 34: *short-directional-svc-phrase*

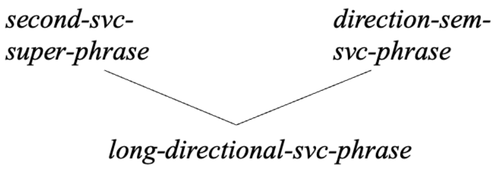


Figure 35: Cross-Inheritance for *long-directional-svc-phrase*

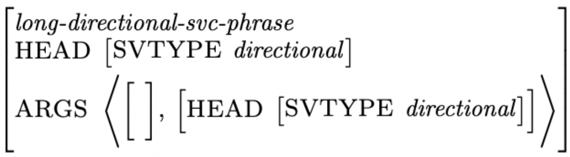


Figure 36: *long-directional-svc-phrase*

In the SVC in (12) then, the VPs *khâam saphaan* ('cross the bridge) and *klàb baan* ('return home') use *short-directional-svc-phrase* (Fig.34) to combine. Next, this SVC combines with the VP *den* ('walk') using *long-directional-svc-phrase* (Fig.36). As *long-directional-svc-phrase* requires VP2 to be headed by a directional verb (*den* ('walk') is a motion verb), this rule cannot be used to combine the resulting VP with any additional verbs. Instead, the final VP combines with the subject *Suri*, forming the sentence in (12).

C Further Information on SVC Interactions

Section 4.3.4 described *deictic-purpose-interact-trans-lex-rule*, which is shown in Fig.37 below:

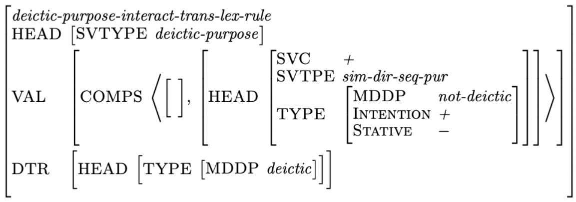


Figure 37: *deictic-purpose-interact-trans-lex-rule*

This lexical rule is used to construct the SVC in (9), which is a Deictic Purpose SVC where VP2 is a Simultaneous SVC. It can also be used to form the SVCs in (13) and (14) below, where VP2 of the Deictic Purpose SVC is a Long Directional or Sequential/Open Purpose SVC respectively.

(13) สุรี ไป วิ่ง ข้าม สะพาน
Suri paj wiŋ khâam saphaan
Suri go run cross bridge
'Suri went to run across the bridge'
(Muansuwan 2002)

(14) สุรี ไป ซื้อ หนังสือ อ่าน
Suri paj sɯ́ɯ nǎŋsɯ́ɯ ʔà:n
Suri go buy book read
'Suri went to buy a book then read (it/the book)'
'Suri went to buy a book to read'

Additional lexical and phrase structure rules for various types of SVC interactions can be found in the thai.tdl file in the implemented grammar.