AN EMPIRICAL COMPARISON OF VERBNET SYNTACTIC FRAMES AND THE SEMLINK CORPUS

by

WESTON RICHARD FEELY

B.A., University of Colorado Boulder, 2012

M.A., University of Colorado Boulder, 2012

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirement for the degree of

Master of Arts

Department of Linguistics

2012

This thesis entitled:

An Empirical Comparison of VerbNet Syntactic Frames and the Semlink Corpus

written by Weston Richard Feely

has been approved for the Department of Linguistics.

_____

Dr. Martha Palmer

_____

Dr. Laura Michaelis-Cummings

Date_____

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

ABSTRACT

Feely, Weston Richard (M.A., Linguistics, Department of Linguistics)

An Empirical Comparison of VerbNet Syntactic Frames and the Semlink Corpus

Thesis directed by Professor Martha Palmer

This paper describes a method of automatically comparing syntactic frames from the verb lexicon VerbNet with syntactic frames from the Semlink corpus. A method of extracting syntactic frames and semantic argument structures is explained, followed by a method of comparing syntactic frames, both directly and by argument structure. The results of the comparison are described in terms of matching success for frame tokens and frame types, divided into categories based on frame type frequency within Semlink. Overall, 54.14% of the frame tokens within Semlink can be directly matched to VerbNet, with an additional 14.32% matching by argument structure. However, only 29.30% of the frame types within Semlink can be matched to VerbNet, suggesting that the comparison method cannot match a majority of the large variation of frames types in Semlink. A set of distinguishing frame types for VerbNet classes is also proposed and included in this work.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Martha Palmer, for her guidance and support not only during this project but also throughout my transition from undergraduate to graduate work; I wouldn't be where I am today without her help. I'd also like to thank my other thesis committee members: Jim Martin and Laura Michaelis-Cummings. Special thanks to Jinho Choi for letting me use his TreeBank API, which I used to retrieve syntactic constituents from the Penn Treebank, and for retrieving the PropBank to VerbNet type-to-type mapping, which I used to fill in missing VerbNet semantic roles for PropBank numbered ARGs. Finally, I'd like to thank my family and friends for their never-ending love and support. Thank you all so much.

CONTENTS

TABLES

Table

FIGURES

Figure

1. Introduction

1.1 General Background

Within the field of computational linguistics, there has been a long history of research on the syntax-semantics interface, in a theoretical approach focused on furthering linguistic knowledge of syntax and semantics using computational methods, and in an applied approach focused on how linguistic knowledge of syntax and semantics can be used to improve natural language processing tasks. Such efforts include creating measures of word meaning based on placement in hierarchical ontologies, like WordNet (Fellbaum 1998), building contextually-based vector space models of word meaning (Erk and Padó 2008), creating information theoretic measures of semantic similarity (Resnik 1995), and automatically labeling syntactic constituents with semantically-relevant thematic roles (Gildea and Jurafsky 2002). In particular, research in computational lexical semantics has sought to further our understanding of how individual words carry meaning within larger syntactic units. It is clear that the predicate argument structure plays a very central role in representing the meaning of a sentence, for a computer. Verbs, in particular, have been of interest to many different computational lexicons.

One such computational lexicon is VerbNet (Dang et al 1998), a hand-crafted verb lexicon that groups English verbs by their syntactic behavior, based on an original verb classification by Levin (1993). Levin proposed that English verbs could be classified according to how a verb and its arguments form syntactic frames, and the syntactic alternations of these frames that a verb allows. VerbNet follows this plan for organizing a large number of English verbs into classes, which list their verb members along with the syntactic frames available to all verbs in the class. VerbNet has been valuable for natural language processing tasks such as detecting event relations (Palmer et al 2009), semantic role labeling (Swier and Stevenson 2005),

and word sense disambiguation (Brown et al 2011).

However, since VerbNet is a largely theoretical resource, little work has been done on comparing the syntactic frames in each VerbNet class with a large corpus of parsed sentences. Semlink (Palmer 2009) is one such corpus of parsed sentences from the Penn Treebank (Marcus et al 1993) that have multiple semantic role annotations from PropBank (Palmer et al 2005), VerbNet (Kipper et al 2008), and FrameNet (Fillmore et al 2003). Each entry in Semlink has a VerbNet classification for the main verb of the sentence, along with VerbNet semantic roles for each constituent in the parse tree that represents the verb's core arguments.

1.2 Project Description

This project is an initial attempt to compare the set of syntactic frames in each VerbNet class to the set of syntactic frames that actually occur in usage in the class's corresponding Semlink entries. Such a comparison is challenging because VerbNet is a largely theoretical verb lexicon which is still strongly rooted in Levin's original classification. Semlink, on the other hand, is an annotated corpus of real language in use, which often shows far more syntactic variability than assumed by theoretical linguistics. Thus, a comparison of VerbNet with Semlink could provide a greater range of syntactic frames for most VerbNet classes, simply because unexpected syntactic frames present themselves in the Semlink data.

This additional syntactic variation in the Semlink data should facilitate the primary goal of this project, which is to make VerbNet a more data-driven lexical resource. The Semlink data will provide a way to empirically validate the class organization of VerbNet by demonstrating which of VerbNet's syntactic frames are present in Semlink data for a given class and which syntactic frames are present in the data that are not listed among the options for a given VerbNet

class.

Additionally, the Semlink data will provide frequency information for syntactic frames, so that each syntactic frame in a VerbNet class can be listed with how often it occurs in corpus data. This is especially important, because our empirical validation of the class organization of VerbNet can be extended to: which syntactic frames are highly frequent in Semlink and present in a given VerbNet class, which frames are highly frequent but missing from a given class, which frames are infrequent and present in a given class, and which frames are infrequent but missing from a given class.

Ultimately, the goals of this project are to: provide an empirical validation for the organization of VerbNet, provide VerbNet with statistical information on the frequency of each class's syntactic frames in the Semlink corpus, and provide data that may prove useful for future work on an empirical clustering of verbs into classes.

1.3 VerbNet Background

Levin (1993) originally proposed a classification for many English verbs based on the alternations of syntactic frames available to the set of verbs in each of her classes. Building on decades of research in the linguistic community, the syntactic alternations available to a Levin class are based on naturally occurring data and native English speaker intuitions about which frames are acceptable for a given verb. One example of a Levin class is the class of "*Cut* verbs," which behave similarly to "cut." This example can be seen in the figure below.

Cut verbs

Class members: chip, clip, cut, hack, hew, saw, scrape, scratch, slash, snip

Properties:

(294) Carol cut the bread with the knife

(295) Conative Alternation:

a. Carol cut the bread.

b. Carol cut at the bread.

(297) *Causative Alternations:

a. Carol cut the bread.

b. *The bread cut.

Figure 1: "*Cut* verbs" from (Levin 1993)

Here, Levin defines the set of verbs that belong to the class of "*Cut* verbs" by listing the verb members of the class and then defining a set of alternations that are or are not acceptable for this class. In the example above, in addition to the standard transitive, "*Cut* verbs" show the conative alternation where the object of a simple transitive instance of the verb as in (294) can appear in a prepositional phrase headed by "at," as in (295.b). However, this class does not allow the causative alternation, where the transitive use of the verb has the meaning of "cause an intransitive instance of the verb." Since the verbs in this class cannot be intransitive, sentences such as (297.b) are ungrammatical and the causative alternation is not available. (Levin 1993)

Similarly, a typical VerbNet (Kipper et al 2008) verb class includes several main parts: a list of the verb members of the class, a list of the semantic roles available for verb arguments, and the syntactic frames that define the class. Unlike the original Levin classes, the syntactic frames in each class are listed explicitly as a linear order of syntactic constituents (NP, PP, etc.) of each verb argument, around the main verb (marked V). These frames are listed along with the semantic role labels for each argument, taken from the list of semantic roles for the class. Finally, a semantic representation is given for each frame, as a set of semantic predicates that describe

the event structure of the frame. This simple VerbNet class, “chase-51.6,” can be seen in Figure 2 below.

No Comments

chase-51.6

Members: 7, Frames: 3

MEMBERS

CHASE (FN 1; WN 1, 2; G 1, 2)
FOLLOW (FN 1; WN 1, 22; G 1)
PURSUE (FN 1; WN 2; G 2)
SHADOW (FN 1; WN 1; G 1)
TAIL (FN 1; WN 1; G 1)
TRACK (FN 1; WN 3; G 2)
TRAIL (FN 1; WN 2; G 3)

ROLES

- AGENT [+ANIMATE]
- THEME [+CONCRETE]
- LOCATION

FRAMES

NP V NP

EXAMPLE "Jackie chased the thief."
SYNTAX AGENT V THEME
SEMANTICS MOTION(DURING(E), AGENT) MOTION(DURING(E), THEME)

NP V NP PP.LOCATION

EXAMPLE "Jackie chased the thief down the street."
SYNTAX AGENT V THEME {{+SPATIAL}} LOCATION
SEMANTICS MOTION(DURING(E), AGENT) MOTION(DURING(E), THEME) PREP(E, THEME, LOCATION) PREP(E, AGENT, LOCATION)

NP V PP.THEME

EXAMPLE "Jackie chased after the thief."
SYNTAX AGENT V {AFTER} THEME
SEMANTICS MOTION(DURING(E), AGENT) MOTION(DURING(E), THEME)

Figure 2: VerbNet Class “chase-51.6”

In the above VN class, there are seven verb members “chase, follow, pursue, shadow, tail, track, trail” which are grouped together by three syntactic frames in which they can appear. The first frame is the simple transitive “NP V NP” frame, with semantic roles “Agent V Theme.” The second frame “NP V NP PP” has an additional prepositional phrase which specifies the location of the action, as specified by the semantic roles for this frame. The last frame “NP V PP” has the

object of the action in a prepositional phrase headed by “after,” and the prepositional phrase takes the role of Theme from the original NP object of the simple transitive frame.

VerbNet classes differ from their original Levin classes, in that there is a greater focus on the frames that a class allows, rather than the syntactic alternations that Levin classes list. For example, the original Levin class in Figure 1 for the VerbNet class in Figure 2 lists the causative alternation as not being available to the class, as in the above Levin class example of “*Cut* verbs.” The fact that this causative alternation is not allowed is a distinguishing feature that separates this Levin class from others. Conversely, the original Levin class mentioned the third “NP V PP” frame only in passing, with the note that it was only available for some of the verbs of this class (Levin 1993). Other than these differences in emphasis – VerbNet placing emphasis on the syntactic frames, Levin classes placing emphasis on the syntactic alternations – this particular VerbNet class is remarkably similar to its original Levin class. This need not be the case. VerbNet classes can have different verb members from the Levin classes they were based on, since VerbNet's classification of verbs is continuing to undergo revision. Also, the VerbNet syntactic frames aren't necessarily an exact replication of the original syntactic alternations from their corresponding Levin classes.

It is also important to mention that VerbNet classes can be considerably more complex than this “chase-51.6” example. VerbNet classes may have multiple levels of representation, where a VerbNet class has its own subclasses. This means that verbs belonging to a subclass have their own set of syntactic frames that are not available to the higher class, in addition to the syntactic frames that the subclass verbs inherit from the higher class. For example, the original Levin class of “*Give* verbs” had a single set of members and syntactic alternations, with the syntactic alternations that were not available to all members of the class noted specifically. The

corresponding VerbNet class, “give-13.1” has instead two levels: the class “13.1” which has a set of members and frames, and its subclass “13.1-1” which has an additional set of members that can take the frames from “13.1” as well as another set of frames specific to the subclass verbs. This more complex class can be seen in Figures 3 and 4 below.

Figure 3: VerbNet Class “give-13.1,” Main Class

The main class for give-13.1 is similar to chase-51.6 from Figure 2, having a set of members: “lend, loan, pass, peddle, refund, render,” a set of semantic roles “Agent, Theme, Recipient,” and a set of frames: “NP V NP PP,” “NP V NP NP,” “NP V NP,” and “NP V PP.” Each frame has a corresponding set of semantic roles in the “Syntax” line under the frame's

constituents. All of the members in this give-13.1 main class have access to the frames from this list, and only the frames in this list.

Figure 4: VerbNet Class “give-13.1,” Subclass

The above Figure 4 is the subclass “give-13.1-1” of give-13.1, marked by a “-1” at the end of the class's number. Like the main class in Figure 3, this subclass has its own set of verb members: “give, hock, lease, pawn, rent, and sell,” its own set of semantic roles, which only includes the new semantic role of “Asset,” and its own set of syntactic frames: “NP V NP PP,” “NP V NP PP PP,” and “NP V NP NP PP.” All the verb members in this subclass have access to all of these frames, and the new semantic role “Asset,” as well as access to all of the frames and semantic roles of the main class in Figure 3. This is an example of the fact that the inheritance in

VerbNet is monotonic; all of the subclasses of a VerbNet class have access to the information in the higher classes they inherit from, but the higher classes do not have access to the information in their subclasses.

1.4 PropBank and FrameNet Background

The other two lexical resources whose annotations are used in Semlink are PropBank (Palmer et al 2005) and FrameNet (Fillmore et al 2003), and they are reviewed here.

1.4.1 PropBank Background

PropBank is a corpus of The Wall Street Journal text from the Penn Treebank (Marcus et al 1993), annotated with predicate-argument relations. These predicate-argument relations are similar to VerbNet's semantic role labels for the arguments of a particular verb in a VerbNet class. PropBank groups lexical information for verbs by verb sense in particular "rolesets." For example, the predicate "chase" has only one roleset, "chase.01," which can be seen in Figure 5 below.

Roleset id: chase.01, *follow, pursue,* vncls: 51.6

Roles:

ARG0: follower (vnrole: 51.6-Agent)

ARG1: thing followed (vnrole: 51.6-Theme)

Example:

"Higher margins would chase away dozens of smaller traders who help larger traders buy and sell, they say."

ARG0: Higher margins, ARGM-MOD: would, Rel: chase, ARGM-DIR: away

ARG1: dozens of smaller traders who help larger traders buy and sell

Figure 5: PropBank Roleset “chase.01”

In the above example, the roleset is “chase.01” which is mapped to the VerbNet class “51.6,” from Figure 2. The semantic roles available to this class are ARG0 and ARG1: two of PropBank's numbered arguments, which are used to label a verb's core arguments. In contrast, PropBank ARGMs are used to label adjuncts to the verb. In the example, ARG0 is the “follower” and ARG1 is the “thing followed.” An example below the role specification for the roleset shows a sentence from PropBank, with its numbered ARG and ARGM labels. This example illustrates that the one frame for “chase” covers both physical usages and abstract usages.

1.4.2 FrameNet Background

FrameNet is another lexical database, which differs from VerbNet and PropBank because FrameNet is based on the linguistic theory of frame semantics. In this theoretical approach, a semantic frame is a description of an event relation and the participants to the event. Unlike VerbNet, the lexical units in FrameNet that evoke a particular semantic frame are not limited to just verbs and the frame elements of a particular FrameNet frame are more fine-grained than VerbNet's semantic roles or PropBank's numbered ARGs and ARGMs. FrameNet is of less importance to this project, since the VerbNet and PropBank annotations are sufficient to compare the syntactic frames in VerbNet with those in Semlink. However, future work could include the use of FrameNet annotations as well, to look for agreement across PropBank, VerbNet, and FrameNet, when empirically validating VerbNet's syntactic frames.

1.5 Semlink Background

Semlink (Palmer 2009) consists of 112,917 sentences of The Wall Street Journal text from the Penn Treebank (Marcus et al 1993), which are all annotated with semantic roles from PropBank (Palmer et al 2005), VerbNet (Kipper et al 2008), and FrameNet (Fillmore et al 2003). Each Semlink instance lists the location of the syntactic parse of the sentence in the Penn Treebank, a PropBank and VerbNet classification for the entry's main verb, the location in the parse of each of the verb's arguments and adjuncts, as well as the semantic roles for each verb argument. An example of a Semlink instance is in Figure 6 below.

wsj/16/wsj_1615.mrg 47 5 auto pursue.01;VN=51.6 vp--a 1:1-ARG0[Agent] 5:0-rel 6:1-ARG1[Theme] 11:1-ARGM-LOC

Figure 6: A Semlink Instance

In this instance, the first three pieces of information specify that the location of the parse is the 47th tree in the file wsj/16/wsj_1615.mrg. In the middle of the instance, the PropBank roleset assignment is specified as “pursue.01” and the VerbNet class assignment for the entry is “VN=51.6,” which is the same “chase-51.6” class from Figure 2. The sentence has two main arguments which are assigned numbered arguments in PropBank (e.g. ARG0, ARG1, ARG2, etc.), an ARG0 which is also annotated with the VerbNet role “[Agent]” and an ARG1 which is also annotated with the VerbNet role “[Theme].” The main verb of the sentence is between these two numbered PropBank ARGs, and it is marked by “rel” for “relation.” The sentence also has one adjunct to the verb, which is labeled with a PropBank ARGM (meaning adjunct): ARGM-LOC, which is a syntactic constituent specifying the location of the event.

Although it is not present in Figure 6 above, the syntactic constituents in the Treebank parse which are labeled with these PropBank and VerbNet annotations are “NP V NP PP.” This

list includes both arguments and adjuncts, however, since PropBank labels the PP as an ARGM-LOC adjunct (the VerbNet annotations list this PP as an adjunct to the verb by simply not giving this constituent a VerbNet semantic role). If we remove the adjunct from the list, we get the syntactic frame: “NP V NP,” which only includes the verb and its core arguments.

However, it's important to note that for this instance (and many others) VerbNet and PropBank don't necessarily agree on what the core arguments of the verb are. Back in Figure 2, the VerbNet class “chase-56.1” lists “NP V NP PP” as an acceptable syntactic frame, with PP as a main argument to the verb. Yet, in this Semlink instance the PP is not labeled with the VerbNet semantic role “[Location]” because in Semlink only PropBank numbered ARGs are given a VerbNet semantic role. This is due to the fact that the VerbNet semantic roles in Semlink are automatically mapped onto only PropBank numbered ARGs. For Semlink, this has the benefit of having a single argument structure representation, by deferring to PropBank's assessment of which constituents are the arguments of the verb every time. However, this will be one of the shortcomings of this project, since this project is an attempt to compare the VerbNet representation of Semlink instances like this to their corresponding frames in the VerbNet class that they are assigned. The frame that we should retrieve from this instance is “NP V NP PP,” according to how VerbNet assesses the arguments of the verb, but since there is no easy method to supply these missing VerbNet roles automatically, we are left with the frame “NP V NP.”

It is important to note that many other Semlink instances have a similar issue, where there is a missing VerbNet role, not for a PropBank ARGM, but a PropBank numbered ARG. This is because of a difference in the argument/adjunct distinction in VerbNet and PropBank: in some cases, VerbNet doesn't consider the PropBank numbered ARG to be a core argument to the verb. This forms the other half of the shortcoming for this project mentioned above, since these

instances will have an additional constituent in the syntactic frame we can retrieve from the instance, which is considered only an adjunct to the verb in VerbNet.

2. Data Extraction

2.1 Semlink Data Extraction

The Semlink data for this project includes 79,815 valid Semlink instances, which are all Semlink instances with a valid VerbNet class assignment. Each of the Semlink instances included in the project data was processed for the necessary information to compare each instance to VerbNet. This included the extraction of each Semlink instance's VerbNet class assignment, the instance's PropBank roleset assignment, the syntactic frame from the Treebank parse, the VerbNet semantic roles for each constituent in the frame, as well as Semlink frequency information for each frame given its VerbNet class. As an example, the data that was extracted from the Semlink instance in Figure 6 is represented as the list in Figure 7 below.

[VN=51.6, pursue.01, NP_V_NP, Agent_V_Theme, 178, 0.76724137931]

Figure 7: Data Extracted from Semlink Instance in Figure 6

The above figure is a list of data automatically extracted from the Semlink instance in Figure 6. The list includes: the VerbNet class assignment for the instance, the PropBank roleset assignment for the instance, the syntactic frame of the main verb and its arguments, the VerbNet semantic role argument structure of the main verb and its arguments, the frequency count of the [VerbNet class, syntactic frame] pair in all of the Semlink instances, and the probability of the [VerbNet class, syntactic frame] pair in all of the Semlink instances. Each of these items is explained in more detail below.

The VerbNet class assignments, the PropBank roleset assignments, and the VerbNet semantic roles were taken directly from the Semlink instances using regular expressions. For the example in Figure 6, the VerbNet class “VN=51.6,” the PropBank roleset “pursue.01,” and the VerbNet semantic roles “Agent” and “Theme” can be seen directly in Figure 6.

The syntactic frames were created using a Penn Treebank API that automatically retrieved the syntactic constituents immediately dominating the part-of-speech tag for each of the words that were marked as arguments to the main verb in the Semlink instances. For example, in Figure 6 the verb arguments are the PropBank numbered ARGs, ARG0 and ARG1. Immediately before each of these PropBank annotations are the markers “1:1” and “6:1.” These markers are the locations in the Penn Treebank sentence's parse of the head nouns of the two constituents that are verb arguments. The constituent immediately above each of these locations was the constituent “NP,” so these constituents were retrieved by the Treebank API. After retrieving the verb argument constituents, they were combined into a flat syntactic frame with the main verb between them: “NP_V_NP.”

The frequency counts were calculated simply by counting the number of times a given VerbNet class assignment and syntactic frame occurred in the same instance. For this instance, the pair [“VN=51.6”, “NP_V_NP”] occurred 178 times. The probabilities were calculated by simply dividing the frequency counts for a given [VerbNet class, frame] pair by the count of all Semlink instances that had the VerbNet class assignment from the pair. The probability in Figure 7 is 76.7%, meaning that 76.7% of the “VN=51.6” instances had a “NP_V_NP” frame.

2.2 VerbNet Data Extraction

The VerbNet data for this project includes the frames and corresponding semantic role argument structures for all VerbNet classes. These frames and argument structures were taken directly from the VerbNet XML files using regular expressions, with some small modifications to each frame. In order to facilitate easier matching with the Semlink frames, the constituents in each of VerbNet's flat syntactic frames were stripped of additional tags, such as: redundant

thematic roles (e.g. PP.location), syntactic alternation tags (e.g. NP-Dative), and other tags (e.g. S_INF).

An example of the data extracted from a VN class is in Figure 8 below.

VN=51.6

[NP_V_NP, Agent_V_Theme]

[NP_V_NP_PP, Agent_V_Theme_Location]

[NP_V_PP, Agent_V_Theme]

Figure 8: Data Extracted from VerbNet Class “chase-51.6”

The data in the above Figure 8 are a simplified description of the VerbNet class from Figure 2. Each entry is a [frame, argument structure] pair, where the constituents in the frame correspond to the semantic role labels in the argument structure.

2.3 Challenges to Data Extraction

In addition to the basic PropBank-VerbNet mismatches mentioned above, there are other potential sources of error. One challenge to the data extraction process, is that there are many Semlink instances with missing VerbNet class assignments, either because there wasn't an agreement on which VerbNet class the main verb of the sentence belongs to or because the main verb of the sentence is not yet assigned to a VerbNet class. Additionally, there are many Semlink instances with VerbNet class assignments that are no longer valid, because the VerbNet class organization has been updated since the time Semlink was created. These instances were ignored because there would be no current VerbNet class to compare them to.

Aside from these ignored instances, there are also many valid Semlink instances which are considered to have erroneous VerbNet class assignments, which are currently being fixed.

Since these instances were not yet fixed at the time this project was completed, these erroneous instances were still included in the data for this project, since this project attempts to compare the current version of Semlink to the current version of VerbNet, despite any errors that may be fixed later. However, this will be considered as another source of error for the project, since these instances formed another significant portion of the available data.

2.3.1 Challenges to Semlink Data Extraction

Although the example in Figure 6 is fairly simple, as stated before in Section 1.5, many other Semlink instances had PropBank numbered ARGs with missing VerbNet semantic roles. The proportion of the Semlink data that had at least one missing VerbNet semantic role for a PropBank numbered ARG was 14.12%. In these cases, the PropBank numbered ARG was retrieved from Semlink instead, leaving many semantic role argument structures that looked like "ARG0_V_ARG1."

However, using the VerbNet class assignments and PropBank roleset assignments, some of these missing VerbNet roles were automatically retrieved from a PropBank ARG to VerbNet semantic role mapping. This extra step brought the proportion of the Semlink data that had at least one missing VerbNet semantic role for a PropBank numbered ARG down to 7.28%. Although this is a great improvement, this 7.28% of the Semlink instances still poses a large source of error for this project, since the VerbNet argument structures that could not be recovered for these instances could have been important to facilitate the matching process by argument structure, described in Section 3 below.

2.3.2 Challenges to VerbNet Data Extraction

The example in Figure 8 above is the same VerbNet class from Figure 2, which is a simple example with no subclasses. For hierarchical VerbNet classes with subclasses, class and subclass were combined into a single list of [frame, argument structure] pairs. This decision to combine each VerbNet class with its subclasses was made because considerable time had passed since Semlink's VerbNet annotations were made. Since the time Semlink was created, VerbNet has changed many subclass classifications for its verb members. For example, in the current version of VerbNet, the class “give-13.1” from Figures 3 and 4 is currently separated into one main class and a single subclass, which can be seen in Figure 9 below.

13.1: {lend, loan, pass, peddle, refund, render}

13.1-1: {give, hock, lease, pawn, rent, sell}

Figure 9: VerbNet Class “give-13.1” Members from Current Version of VerbNet

However, the Semlink 13.1 instances reveal an older version of VerbNet's organization for this class, which can be seen in Figure 10 below.

13.1: {loan, pass, render}

13.1-1: {give, lease, lend, loan, pass, pawn, pay, peddle, refund, render, rent, repay, retail, sell}

13.1-2: {feed}

Figure 10: VerbNet Class “give-13.1” Members from Semlink

This difference shows that VerbNet has moved members between subclasses, removed members from the class, added new members to the class, and removed an entire subclass since Semlink was created. All of these changes also affect the distribution of frames between the subclasses, since each subclass has a set of frames only available to that subclass. If the VerbNet class and subclass were not combined, many frames in Semlink would not have matched their

corresponding VerbNet class during the matching process in Section 3, simply because the frame had been moved into a different subclass since Semlink was created.

3. Comparison of Semlink frames with VerbNet

After extracting the data from Semlink and VerbNet, the data from each Semlink instance was matched against the set of [frame, argument structure] pairs in the corresponding VerbNet class. This matching process was done using regular expressions in a three step process. First, the frame from the Semlink instance was checked against each of the frames in its corresponding VerbNet class. If there was a match, the instance was counted as having matched a VerbNet frame and if the [VerbNet class, frame] pair for this Semlink instance had not previously been matched, it was added to a list of frame types that matched VerbNet. Second, if the frame from the Semlink instance did not match any of the frames in the corresponding VerbNet class, then the argument structure for the instance was checked against each of the argument structures in the corresponding VerbNet class. If there was a match, the instance was counted as having matched a VerbNet frame and if the [VerbNet class, frame] pair for the Semlink instance had not previously been matched, it was added to a different back-off list of frame types that matched VerbNet. Third, if the frame and its argument structure from the Semlink instance did not match any of the frames in the corresponding VerbNet class, then if the [VerbNet class, frame] pair for the Semlink instance had not previously failed to match, it was added to a final list of frame types that did not match VerbNet.

The end result of this matching process was three counters and three lists. The counters are the portion of the total Semlink instances that matched a VerbNet frame, did not match a frame but did match a VerbNet argument structure, or did not match VerbNet at all. These token counters were converted into token percentages in Table 1 in Section 4 below. The lists contain frame types for each matching condition: frame types that were in VerbNet, frame types that had argument structures that were in VerbNet, and frame types that were not in VerbNet. These type

lists were converted into type percentages in Table 3 in Section 4 below.

This matching process was repeated for three frequency subdivisions of the Semlink frame types: high frequency, middle frequency, and low frequency. These frequency categories were defined as the top 30%, middle 40%, and bottom 30% of the Semlink frame types for each VerbNet class, ranked by frequency. For this second matching process using frequency information, the Semlink frames that matched VerbNet by frame and by argument structure were combined into one category of frame types that matched VerbNet. The Semlink frames that did not match VerbNet by frame or argument structure were left in a separate category of frame types that did not match VerbNet. In the same manner as the first matching process, the end result was a set of counters for the frame tokens that matched VerbNet, and a set of lists for the frame types that matched VerbNet, subdivided by these frequency categories. The percentages of the Semlink frame tokens for each of these frequency subdivisions are in Table 2 of Section 4 below, and the percentages of the Semlink frame types for each of these frequency subdivisions are in Table 4 of Section 4 below.

4. Results

The results of the Semlink token to VerbNet frame matching process described in Section 3 are in Table 1, below.

Percentage of Semlink tokens that...	% of total Semlink tokens (79815)
Matched a VN Frame	54.14%
Matched a VN Argument Structure	14.32%
Did not match corresponding VN class	31.53%

Table 1: Results of Matching Process for Semlink Frame Tokens

Of the 79,815 valid Semlink instances, 54.14% had a frame that matched the corresponding VerbNet class, while 14.32% did not match a frame but did match one of the argument structures in the corresponding VerbNet class. Finally, 31.53% of the Semlink instances did not match a frame or argument structure in the corresponding VerbNet class.

These frame token percentages can also be further divided by the Semlink frequency of the [VerbNet class, frame] pair for each token, which can be seen in the table below.

Match/No match grouping	Frequency	% of total Semlink frame tokens
Matched VerbNet	High Frequency (top 30%)	55.60%
	Middle Frequency (middle 40%)	12.55%
	Low Frequency (bottom 30%)	0.31%
Did not match VerbNet	High Frequency (top 30%)	22.93%
	Middle Frequency (middle 40%)	7.52%
	Low Frequency (bottom 30%)	1.09%

Table 2: Results of Matching Process for Semlink Frame Tokens, Divided by Frequency

In Table 2 above, those tokens that matched a VerbNet frame and those tokens that did not match a VerbNet frame but did match a VerbNet argument structure are combined into a single grouping of frames that matched VerbNet. The frames that matched neither a VerbNet

frame nor a VerbNet argument structure are in a separate grouping of frames that did not match VerbNet. The tokens were divided into frequency categories of high frequency, middle frequency, and low frequency. The divisions between these three categories were tokens whose frame types were in the top 30%, middle 40%, and bottom 30% of their VerbNet class, respectively.

The percentages in Tables 1 and 2 describe portions of all Semlink instances, which is to say they represent frame tokens, not frame types. Since a small number of syntactic frames types are highly frequent in the data (for example, “NP V NP” constitutes 30% of all Semlink instances), it's helpful to also consider the same percentages for frame types, which are in the tables below.

Percentage of Semlink frame types that...	% of total Semlink frame types (3518)
Matched a VN Frame	14.75%
Matched a VN Argument Structure	14.55%
Did not match corresponding VN class	70.69%

Table 3: Results of Matching Process for Semlink Frame Types

The results of the Semlink frame type to VerbNet frame matching process are in Table 3, above. Of the 3,518 Semlink frame types, 14.75% were matched to a VerbNet frame in the corresponding class and 14.55% did not match a VerbNet frame but did match a VerbNet argument structure in the corresponding class. The remaining 70.69% of the Semlink frame types did not match their corresponding VerbNet class at all, meaning 70.69% of the various Semlink frame types were unaccounted for in VerbNet.

These frame type percentages can also be further divided by the Semlink frequency of the [VerbNet class, frame] pair for each type, which can be seen in the table below.

Match/No match grouping	Frequency	% of total Semlink frame types
Matched VerbNet	High Frequency (top 30%)	14.35%
	Middle Frequency (middle 40%)	9.24%
	Low Frequency (bottom 30%)	5.71%
Did not match VerbNet	High Frequency (top 30%)	16.97%
	Middle Frequency (middle 40%)	31.07%
	Low Frequency (bottom 30%)	22.65%

Table 4: Results of Matching Process for Semlink Frame Types, Divided by Frequency

Table 4 is formatted like Table 2, above, and it shows that most of the Semlink frame types did not match VerbNet and were in the middle frequency category (31.07%) and the low frequency category (22.65%). Of the Semlink frame types that did match VerbNet, most were in the high frequency category (14.35%).

5. Discussion

The results presented in Table 1 above demonstrate that although over half (54.14%) of the Semlink instances can be matched to VerbNet using their syntactic frames alone, an additional 14.32% of the instances can be matched to VerbNet using the semantic roles of each verb argument. The importance of these additional frames is even greater, considering that only 14.75% of the total Semlink frame types matched VerbNet directly, but an additional 14.55% of the total frame types matched by argument structure. This is an indication that nearly twice as much of the variation of frame types can be recovered when matching Semlink to VerbNet, by using the semantic role labels of each verb argument.

Many of the frame types retrieved using an argument structure match include Penn Treebank variants of the sentence phrase "S," such as "SBAR," "SBARQ," "SQ," and "SINV." For example, the frame "NP V S" is present in the VerbNet class "say-37.7," this pair from Semlink matched VerbNet by frame. However, there were far more Semlink instances of "NP V SBAR" for this VerbNet class, and this pair was matched by argument structure, since none of the VerbNet frames for this class include "SBAR." This may be an indication that a future version of this project should ignore variants of phrasal categories like "S," so these variants can be matched directly, or this may simply be another reason why the argument structure matches were an important part of the matching process.

However, while the additional frame type matches may include many frame type variants that should be in their respective VerbNet classes, since the argument structures were the same as frames already present in VerbNet, it may also be the case that many of the frame types in this 14.55% are errors. Some of the frame types that make up this 14.55% include peculiar frames such as PP-initial frames and frames with part-of-speech tags as arguments rather than a phrase,

which may be errors that should be looked at individually in future work.

5.1 Frame Matches by Frequency

5.1.1 High Frequency Matches

Most of the Semlink instances that matched VerbNet were in the high frequency category (55.60%), although many of these frames were the simple transitive “NP V NP,” which is especially common across all of Semlink, making up 30% of all Semlink instances, alone. This is supported by the fact that the high frequency match portion for frame types makes up only 14.35% of the frame types. This is a stark difference from the 55.60% of the matching frame tokens for this frequency category, which is an indication that frames that are highly common across all of Semlink skew the frame token data, which is why it is important to consider the frame types percentages in Tables 3 and 4, when doing analysis of the Semlink frames missing from VerbNet.

The high frequency Semlink frame types that matched VerbNet make up 14.35% of the total Semlink frame types, which is the largest percentage of the frame types that matched VerbNet. This means that among the frame types that matched VerbNet, the greatest amount of frame type variation was in the high frequency range (top 30%) for each VerbNet class. This is an encouraging result, since a VerbNet class's most frequent frames should be the ones that are best represented in Semlink.

This is also an important category of the frame type matches, since it includes a portion of frame type matches that could be highly useful for future work on automatic verbs clustering. Frames that are high frequency in a particular class but low frequency across all of Semlink were expected to be highly distinguishing frames for that class. Of the 505 high frequency Semlink

frame types that matched VerbNet, 72.28% had a higher frequency within their VerbNet class than their frequency across all of Semlink, which may be a good starting point for determining which frames are highly distinguishing of their respective VerbNet classes. This list of distinguishing frames can be seen in the appendix.

5.1.2 Middle Frequency Matches

The next largest percentage of the Semlink frame tokens (12.55%) and frame types (9.24%) that matched VerbNet is the middle frequency category, which makes up the middle 40% of each VerbNet class's frame types. Since the middle frequency matches are neither quite frequent nor infrequent, this portion of the Semlink frame types may not particularly useful for future work. However, this portion may prove to be a useful portion of the frame types for improving the frame matching process in future versions of this project, because many of the frame types in this category are the possibly erroneous frames from the argument structure matches, such as the PP-initial frames, and the frames that have part-of-speech tags instead of phrasal constituents.

5.1.3 Low Frequency Matches

Finally, the smallest percentage of the Semlink frame tokens (0.31%) and frame types (5.71%) that matched VerbNet were in the low frequency category, which makes up the bottom 30% of each VerbNet class's frame types. While the high and middle frequency categories have a larger frame type percentage than frame token percentage, indicating that high and middle frequency frames skew the data towards their respective categories, the low frequency category has a much lower frame token (0.31%) than frame types (5.71%) percentage. This is probably

because this low frequency match category is almost entirely made up of very few, but different, frame tokens. Most of the frame types in this category occurred in just a single Semlink instance. These frame matches are good support for why VerbNet should include frequency information, since there is currently no indication of which frames in a VerbNet class occur quite often, and which have only a single matching Semlink instance.

However, many of these single instance frame type matches may also be errors from the argument structure matching process, like many of the frame types in the middle frequency category. If this is the case, then this category may also be useful for improving the frame types matching process in future versions of this project.

5.2 Gaps in the Mapping Process

Aside from these Semlink frame types that can be matched to VerbNet through frame (14.75%) or argument structure (14.55%), the majority of the Semlink frame types (70.69%) cannot be matched to VerbNet at all, which may either show a deficiency in the types of syntactic frames in VerbNet, or a failure of this method of matching Semlink instances to VerbNet.

However, looking over the list of these frames types reveals that many of the frame types that did not match VerbNet are verb-initial frames, such as “V NP PP” (compare with “NP V NP PP”). These verb-initial frames make up 50.42% of the frame types that did not match VerbNet, as well as 18.58% of the total Semlink instances. Only 1.47% of the frame types retrieved from VerbNet were verb-initial, and most of these were retrieved from VerbNet without the frame's initial word such as “There” in “There V NP PP” (from the class “meander-47.7,” among others). However, the majority (98.53%) of the frames retrieved from VerbNet were not verb-initial, which made matching the verb-initial frames in Semlink unlikely.

It may be the case that these verb-initial frames represent Penn Treebank parses with a null subject or trace that was not included in the parse during treebanking, and thus could not be annotated with a PropBank and VerbNet argument label. If these verb-initial frames are seen as variants of frames with NP-subjects, such as “V NP PP” being a variant of “NP V NP PP,” then the percentage of frame types that did not match VerbNet may be as low as 35.05%. However, this is impossible to say for sure, without further work examining the Penn Treebank parses for the 18.58% of all Semlink instances that had frames that were verb-initial.

Among the other frames that were not matched to VerbNet, there were some oddities such as a part-of-speech tag being retrieved instead of a phrase (“DT” in “DT V NP”) which may be due to a treebanking error, or an error in retrieving the right level of the phrase in the Treebank parse. Also, there were many verb-particle frames (those with the part-of-speech tag “PRT” in the frame) that did not match their respective VerbNet classes. Although these verb-particle frames made up only 2.94% of the frame types that did not match VerbNet, this may be an important portion of the frame types that failed to match, given that half of the frame types that failed to match VerbNet were verb-initial frames. Additionally, there were many non-verb-initial frames (10.98%) with an adverbial phrase or adjective phrase that did not match, but otherwise do not appear to be erroneous. These may be important variations on frames already in VerbNet, and should be examined in future work, as well.

6. Conclusions and Future Work

Overall, the frame matching process described in this work seems to be valuable for assessing frame types that occur in Semlink and are given a VerbNet class assignment, but are missing from their respective VerbNet classes. However, the results of matching frame tokens (i.e. percentages of Semlink instances) do not seem to be useful for this same analysis, since a small number of highly frequent frame types skew the frame token data. This work has highlighted particular frame types that might be useful additions to VerbNet, based on the Semlink data, across categories of the matching process described here. These included frame types that matched VerbNet through the frame's semantic roles, which have more specific Penn Treebank variants of the phrasal constituents in VerbNet frames (such as “SBAR” in place of “S”). These also include frame types that did not match VerbNet, but may still be useful additions to future versions of VerbNet, such as the verb-initial Semlink frame types, the verb-particle constructions that did not match, and frame types that had an additional adverbial phrase or adjective phrase included as an argument to the verb.

Future work on this particular method of comparing VerbNet and Semlink will include an assessment of the many verb-initial frames found in the Semlink instances, focusing on why these frames made up 18.58% of the total Semlink instances and how these frames might be matched to VerbNet in future versions of this project. Also, the strange errors in the frame retrieval will be examined, such as the frames with part-of-speech tags in place of their higher phrasal constituents, unexpected PP-initial frames, and other frames that are highly different from the expected VerbNet set of frames in a given class.

Future work should also include a close examination of the Semlink frame types that matched VerbNet and were part of the top 30% of their VerbNet class, but were lower frequency

across all of Semlink, since these frames may prove to be distinguishing frames for these VerbNet classes. Such distinguishing frames can be used theoretically, in a comparison of what differentiates similar VerbNet classes, but also empirically, as features for an unsupervised clustering of verbs into classes similar to VerbNet. This, in particular, would further the goal of this project of working towards data-driven semantics and empirical methods for linguistic analysis, since VerbNet remains a largely theoretical lexical resource, which is still highly rooted in its original Levin classification. VerbNet's classification may be too constrained by its original Levin classes, and would benefit from further comparison with an empirical clustering of verbs, based on syntactic and semantic features. Hopefully, syntactic frames types from this project could be used as these syntactic features for future work on this kind of unsupervised verb clustering.

The following table summarizes the suggestions for future work made in this section and provides a possible prioritization.

1	Correction of errors in frame retrieval
2	Examination of verb-initial frames retrieved from Semlink
3	Examination of variant frames from Semlink that did not match VerbNet
4	Examination of possible distinguishing Semlink frame types that matched VerbNet
5	Automatic verb clustering using highly distinguishing VerbNet syntactic frames

Table 5: Suggestions for Future Work

BIBLIOGRAPHY

Brown, Susan W., Dmitriy Dligach, and Martha Palmer. "VerbNet Class Assignment as a WSD Task." *In IWSC 2011: Proceedings of the 9th International Conference on Computational Semantics* (2011).

Dang, Hoa Trang, et al. "Investigating regular sense extensions based on intersective Levin classes." *COLING/ACL-98, 36th Annual Meeting of the Association for Computational Linguistics* (1998): 293-300.

Erk, Katrin, and Sebastian Padó. "A structured vector space model for word meaning in context." *In Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, Honolulu, HI, USA.* (2008).

Fellbaum, Christiane. *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, 1998.

Fillmore, Charles J., Christopher R. Johnson, and Miriam R. L. Petruck. "Background to FrameNet." *International Journal of Lexicography* 16.3 (2003): 235-250.

Gildea, Daniel, and Daniel Jurafsky. "Automatic labeling of semantic roles." *Computational Linguistics* 28.3 (2002).

Kipper, Karin, et al. "A Large-scale Classification of English Verbs." *Language Resources and Evaluation* 42.1 (2008): 21-40.

Levin, Beth. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, 1993.

Marcus, Mitchell, Beatrice Santorini, and Mary Ann Marcinkiewicz. "Building a Large Annotated Corpus of English: The Penn Treebank." *Computational Linguistics* 19.2 (1993): 257-285.

Palmer, Martha, et al. "Leveraging lexical resources for the detection of event relations." *Proceedings of the AAAI* (2009).

Palmer, Martha, Daniel Gildea, and Paul Kingsbury. "The Proposition Bank: An Annotated Corpus of Semantic Roles." *Computational Linguistics* 31.1 (2005): 71-105.

Palmer, Martha. "Semlink: Linking PropBank, VerbNet, and FrameNet." *Proceedings of the Generative Lexicon Conference, GenLex-09* (2009).

Resnik, Philip. "Using information content to evaluate semantic similarity." *Proceedings of IJCAI 14, Montreal, Canada* (1995).

Swier, Robert S., and Suzanne Stevenson. "Exploiting a Verb Lexicon in Automatic Semantic Role Labelling." *Proceedings of the Joint Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP-05)* (2005): 883-890.

APPENDIX

A. List of Distinguishing Frames by VerbNet Class

VerbNet Class	Frame	Count in Class	Frequency in Class	Frequency across All of Semlink
10.1	NP_V_NP	207	0.543307087	0.304779803
10.1	NP_V_NP_PP	53	0.139107612	0.066929775
10.1	NP_V_PP	25	0.065616798	0.065113074
10.2	NP_V_NP	27	0.529411765	0.304779803
10.3	NP_V_NP	22	0.628571429	0.304779803
10.3	NP_V_NP_PP	6	0.171428571	0.066929775
10.4.1	NP_V_NP	50	0.434782609	0.304779803
10.4.1	NP_V_NP_PP	19	0.165217391	0.066929775
10.4.2	NP_V_NP	14	0.482758621	0.304779803
10.4.2	NP_V_NP_PP	5	0.172413793	0.066929775
10.5	NP_V_NP	289	0.60460251	0.304779803
10.5	NP_V_NP_PP	77	0.161087866	0.066929775
10.6	NP_V_NP	56	0.335329341	0.304779803
10.6	NP_V_NP_PP	47	0.281437126	0.066929775
10.7	NP_V_NP	22	0.423076923	0.304779803
11.1	NP_V_NP	389	0.46035503	0.304779803
11.1	NP_V_NP_PP	131	0.155029586	0.066929775
11.1	NP_V_NP_NP	21	0.024852071	0.010574453
11.1	NP_V_NP_PP_PP	19	0.022485207	0.006139197
11.1	NP_V_NP_VP	14	0.016568047	0.004372612
11.1	NP_V_PP_PP	11	0.013017751	0.006013907
11.1	NP_V_PP_NP	8	0.009467456	0.001328071
11.2	NP_V	28	0.314606742	0.089782622
11.3	NP_V_NP	105	0.307917889	0.304779803
11.3	NP_V_NP_PP	102	0.299120235	0.066929775
11.3	NP_V_PP_NP	11	0.032258065	0.001328071
11.3	NP_V_NP_ADVP	11	0.032258065	0.002004636
11.3	NP_V_NP_NP	6	0.017595308	0.010574453
11.4	NP_V_NP	50	0.510204082	0.304779803
11.4	NP_V_NP_PP	13	0.132653061	0.066929775
11.4	NP_V_NP_PRT	3	0.030612245	0.000325753
11.4	NP_V_PRT_NP	3	0.030612245	0.001064963
11.5	NP_V_NP_PP	18	0.18	0.066929775
12	NP_V_NP	74	0.411111111	0.304779803

12	NP_V_PP	35	0.194444444	0.065113074
13.1	NP_V_NP	1039	0.362273361	0.304779803
13.1	NP_V_NP_PP	439	0.15306834	0.066929775
13.1	NP_V_NP_NP	381	0.132845188	0.010574453
13.1	NP_V_NP_PP_PP	48	0.016736402	0.006139197
13.1	NP_V_NP_NP_PP	9	0.003138075	0.000300695
13.1	NP_V_PP_NP	7	0.002440725	0.001328071
13.2	NP_V_NP	161	0.305502846	0.304779803
13.2	NP_V_PP	152	0.288425047	0.065113074
13.2	NP_V_NP_PP	81	0.15370019	0.066929775
13.3	NP_V_NP	631	0.386405389	0.304779803
13.3	NP_V_NP_PP	152	0.09308022	0.066929775
13.3	NP_V_NP_NP	76	0.04654011	0.010574453
13.4.1	NP_V_NP	299	0.4784	0.304779803
13.4.1	NP_V_NP_PP	171	0.2736	0.066929775
13.4.1	NP_V_NP_NP	18	0.0288	0.010574453
13.4.2	NP_V_PP	56	0.312849162	0.065113074
13.4.2	NP_V_NP_PP	36	0.201117318	0.066929775
13.5.1	NP_V_NP	2019	0.693337912	0.304779803
13.5.1	NP_V_NP_PP	279	0.09581044	0.066929775
13.5.1	NP_V_NP_VP	19	0.006524725	0.004372612
13.5.1	NP_V_NP_ADVP	7	0.002403846	0.002004636
13.5.2	NP_V_NP	849	0.606428571	0.304779803
13.5.2	NP_V_NP_PP	225	0.160714286	0.066929775
13.6	NP_V_PP	61	0.125256674	0.065113074
13.6	NP_V_NP_PP	61	0.125256674	0.066929775
14	NP_V_NP	122	0.532751092	0.304779803
14	NP_V_SBAR	36	0.15720524	0.114514816
14	NP_V_PP	16	0.069868996	0.065113074
15.1	NP_V_NP	353	0.522189349	0.304779803
15.2	NP_V_NP	96	0.395061728	0.304779803
15.2	NP_V_NP_PP	73	0.300411523	0.066929775
15.2	NP_V_PP	24	0.098765432	0.065113074
16	NP_V_NP	37	0.627118644	0.304779803
17.1	NP_V_NP	119	0.416083916	0.304779803
17.1	NP_V_NP_PP	37	0.129370629	0.066929775
17.2	NP_V_NP_PP	1	0.333333333	0.066929775
18.1	NP_V_NP	80	0.457142857	0.304779803
18.1	NP_V_NP_PRT	3	0.017142857	0.000325753
18.2	NP_V_NP	6	0.461538462	0.304779803
18.3	NP_V_NP	8	0.444444444	0.304779803

18.4	NP_V	24	0.292682927	0.089782622
19	NP_V_NP	5	0.714285714	0.304779803
20	NP_V_NP	13	0.541666667	0.304779803
21.1	NP_V_NP	164	0.495468278	0.304779803
21.1	NP_V_NP_PP	53	0.160120846	0.066929775
21.1	NP_V_PP	30	0.090634441	0.065113074
21.1	NP_V_NP_PP_PP	26	0.078549849	0.006139197
21.2	NP_V_NP	33	0.559322034	0.304779803
22.1	NP_V_NP	259	0.358725762	0.304779803
22.1	NP_V_NP_PP	164	0.227146814	0.066929775
22.1	NP_V_PP	74	0.102493075	0.065113074
22.2	NP_V_NP	281	0.350374065	0.304779803
22.2	NP_V_PP	177	0.220698254	0.065113074
22.2	NP_V_NP_PP	60	0.074812968	0.066929775
22.3	NP_V_NP_PP	21	0.256097561	0.066929775
22.3	NP_V_PP	7	0.085365854	0.065113074
22.4	NP_V_NP_PP	22	0.173228346	0.066929775
22.4	NP_V_PP	21	0.165354331	0.065113074
23.1	NP_V_NP_PP	10	0.161290323	0.066929775
23.2	NP_V_NP_PP	29	0.145728643	0.066929775
23.2	NP_V_PP	24	0.120603015	0.065113074
23.3	NP_V_NP	17	0.62962963	0.304779803
23.3	NP_V_PP	3	0.111111111	0.065113074
23.4	NP_V	1	0.333333333	0.089782622
25.1	NP_V_NP	128	0.581818182	0.304779803
25.1	NP_V_NP_PP	17	0.077272727	0.066929775
25.2	NP_V_NP	99	0.452054795	0.304779803
25.3	NP_V_NP	24	0.338028169	0.304779803
25.4	NP_V_NP	49	0.480392157	0.304779803
26.1	NP_V_NP	652	0.652	0.304779803
26.2	NP_V_NP	58	0.522522523	0.304779803
26.2	NP_V_NP_PP	10	0.09009009	0.066929775
26.4	NP_V_NP	476	0.49122807	0.304779803
26.5	NP_V_PP	23	0.403508772	0.065113074
26.6.1	NP_V_PP	9	0.310344828	0.065113074
26.6.2	NP_V_PP	150	0.262237762	0.065113074
26.6.2	NP_V_NP_PP	45	0.078671329	0.066929775
26.7	NP_V_NP	175	0.550314465	0.304779803
26.7	NP_V	70	0.220125786	0.089782622
27	NP_V_NP	279	0.661137441	0.304779803
27	NP_V_S	63	0.1492891	0.08370607

29.1	NP_V_NP	615	0.511647255	0.304779803
29.1	NP_V_S	107	0.089018303	0.08370607
29.1	NP_V_NP_NP	26	0.021630616	0.010574453
29.1	NP_V_NP_VP	7	0.005823627	0.004372612
29.2	NP_V_NP	708	0.445843829	0.304779803
29.2	NP_V_NP_PP	204	0.128463476	0.066929775
29.2	NP_V_S	140	0.088161209	0.08370607
29.3	NP_V_NP_NP	154	0.200520833	0.010574453
29.3	NP_V_NP_ADJP	56	0.072916667	0.002154983
29.3	NP_V_NP_VP	6	0.0078125	0.004372612
29.4	NP_V_SBAR	856	0.594031922	0.114514816
29.4	NP_V_S	132	0.091603053	0.08370607
29.4	NP_V_PP_PP	12	0.00832755	0.006013907
29.4	NP_V_VP	9	0.006245663	0.002480737
29.5	NP_V_SBAR	625	0.344352617	0.114514816
29.5	NP_V_S	280	0.154269972	0.08370607
29.5	NP_V_VP	10	0.005509642	0.002480737
29.6	NP_V_PP	126	0.490272374	0.065113074
29.7	NP_V_NP	9	0.333333333	0.304779803
29.8	NP_V_NP	61	0.61	0.304779803
29.8	NP_V_PP	7	0.07	0.065113074
29.9	NP_V_S	68	0.607142857	0.08370607
30.1	NP_V_NP	433	0.495990836	0.304779803
30.1	NP_V_S	139	0.159221077	0.08370607
30.1	NP_V_ADJP	49	0.056128293	0.006790704
30.1	NP_V_VP	12	0.013745704	0.002480737
30.2	NP_V_NP	171	0.53271028	0.304779803
30.3	NP_V_PP	28	0.528301887	0.065113074
30.4	NP_V_ADJP	21	0.428571429	0.006790704
30.4	NP_V_PP	14	0.285714286	0.065113074
30.4	NP_V_ADVP	2	0.040816327	0.006289545
31.1	NP_V_NP	546	0.434367542	0.304779803
31.2	NP_V_NP	389	0.661564626	0.304779803
31.2	NP_V_S	57	0.096938776	0.08370607
31.3	NP_V_PP	155	0.1843044	0.065113074
31.4	NP_V_PP	26	0.325	0.065113074
31.4	NP_V	15	0.1875	0.089782622
32.1	NP_V_S	542	0.536633663	0.08370607
32.1	NP_V_NP	335	0.331683168	0.304779803
32.2	NP_V_S	124	0.494023904	0.08370607
32.2	NP_V_SBAR	98	0.390438247	0.114514816

33	NP_V_NP	214	0.466230937	0.304779803
33	NP_V_NP_PP	104	0.226579521	0.066929775
35.1	NP_V_NP	4	0.5	0.304779803
35.2	NP_V_NP	35	0.346534653	0.304779803
35.2	NP_V_PP	30	0.297029703	0.065113074
35.3	NP_V_NP	27	0.627906977	0.304779803
35.4	NP_V_NP	171	0.737068966	0.304779803
36.1	NP_V_PP	136	0.334975369	0.065113074
36.1	NP_V	69	0.169950739	0.089782622
36.1	NP_V_PP_PP	21	0.051724138	0.006013907
36.2	NP_V_NP	12	0.705882353	0.304779803
36.3	NP_V_NP	130	0.440677966	0.304779803
36.3	NP_V_PP	69	0.233898305	0.065113074
36.3	NP_V	65	0.220338983	0.089782622
36.4	NP_V	5	0.106382979	0.089782622
37.1.1	NP_V_NP	21	0.512195122	0.304779803
37.1.1	NP_V_NP_PP	6	0.146341463	0.066929775
37.1.1	NP_V_S	4	0.097560976	0.08370607
37.1	NP_V_SBAR	42	0.575342466	0.114514816
37.1	NP_V_S	11	0.150684932	0.08370607
37.11	NP_V_PP	83	0.474285714	0.065113074
37.11	NP_V	79	0.451428571	0.089782622
37.2	NP_V_NP_SBAR	133	0.377840909	0.002706258
37.2	NP_V_NP_S	110	0.3125	0.003407881
37.2	NP_V_NP_NP	15	0.042613636	0.010574453
37.3	NP_V_S	12	0.12371134	0.08370607
37.3	NP_V	11	0.113402062	0.089782622
37.3	NP_V_PP	10	0.103092784	0.065113074
37.4	NP_V_NP	93	0.481865285	0.304779803
37.4	NP_V_SBAR	23	0.119170984	0.114514816
37.4	NP_V_NP_PP	19	0.098445596	0.066929775
37.5	NP_V_PP	108	0.534653465	0.065113074
37.5	NP_V	40	0.198019802	0.089782622
37.5	NP_V_PP_PP	8	0.03960396	0.006013907
37.7	PP_V_S	1	8.21E-05	7.52E-05
37.7	PRP$_V_NNS	1	8.21E-05	1.25E-05
37.7	NP_V_X	1	8.21E-05	5.01E-05
37.7	ADVP_V_SBAR	1	8.21E-05	1.25E-05
37.7	NP_V_PRN	1	8.21E-05	7.52E-05
37.7	SBAR_V_SBAR	1	8.21E-05	7.52E-05
37.8	NP_V_SBAR	45	0.340909091	0.114514816

37.8	NP_V_PP	42	0.318181818	0.065113074
37.8	NP_V	12	0.090909091	0.089782622
37.9	NP_V_SBAR	57	0.176470588	0.114514816
37.9	NP_V_NP_PP	39	0.120743034	0.066929775
37.9	NP_V_NP_SBAR	38	0.117647059	0.002706258
37.9	NP_V_NP_VP	35	0.108359133	0.004372612
37.9	NP_V_NP_S	22	0.068111455	0.003407881
38	NP_V	2	0.666666667	0.089782622
39.1	NP_V_NP	30	0.576923077	0.304779803
39.1	NP_V	14	0.269230769	0.089782622
39.2	NP_V_NP	6	0.545454545	0.304779803
39.3	NP_V_NP	8	0.666666667	0.304779803
39.4	NP_V_NP	10	0.454545455	0.304779803
39.5	NP_V_PP	2	0.5	0.065113074
40.1.2	NP_V	3	0.5	0.089782622
40.2	NP_V	26	0.433333333	0.089782622
40.2	NP_V_PP	15	0.25	0.065113074
40.3.1	NP_V_PP	28	0.666666667	0.065113074
40.3.1	NP_V	5	0.119047619	0.089782622
40.3.2	NP_V_NP	16	0.666666667	0.304779803
40.3.3	NP_V_PP	4	0.5	0.065113074
40.4	NP_V	13	0.8125	0.089782622
40.5	NP_V	20	0.408163265	0.089782622
40.5	NP_V_PP	17	0.346938776	0.065113074
40.7	NP_V_NP	10	0.555555556	0.304779803
40.7	NP_V_PP	2	0.111111111	0.065113074
40.8.2	NP_V_PP	4	0.333333333	0.065113074
40.8.2	NP_V	3	0.25	0.089782622
40.8.3	NP_V_NP	22	0.349206349	0.304779803
41.1.1	NP_V	12	0.631578947	0.089782622
41.2.1	NP_V_NP_PP	1	0.5	0.066929775
41.3.1	NP_V_NP	50	0.961538462	0.304779803
41.3.2	NP_V_NP	3	0.75	0.304779803
42.1	NP_V_NP	83	0.592857143	0.304779803
42.2	NP_V_NP	5	0.3125	0.304779803
43.1	NP_V	14	0.5	0.089782622
43.1	NP_V_PP	4	0.142857143	0.065113074
43.2	NP_V	35	0.416666667	0.089782622
43.2	NP_V_PP	15	0.178571429	0.065113074
43.3	NP_V_PP	2	0.5	0.065113074
43.4	NP_V_NP	29	0.337209302	0.304779803

43.4	NP_V	15	0.174418605	0.089782622
44	NP_V_NP	67	0.478571429	0.304779803
45.1	NP_V_NP	41	0.525641026	0.304779803
45.1	NP_V	18	0.230769231	0.089782622
45.2	NP_V	4	0.444444444	0.089782622
45.3	NP_V	10	0.625	0.089782622
45.4	NP_V	1260	0.351366425	0.089782622
45.4	NP_V_NP	1204	0.335750139	0.304779803
45.4	NP_V_ADVP	34	0.009481316	0.006289545
45.4	S_V_NP	9	0.00250976	0.001240368
45.4	PP_V	8	0.002230898	0.000789325
45.5	NP_V	36	0.371134021	0.089782622
45.6	NP_V	517	0.188548505	0.089782622
45.6	NP_V_PP	216	0.078774617	0.065113074
45.6	NP_V_ADVP	113	0.041210795	0.006289545
47.1	NP_V	242	0.221611722	0.089782622
47.1	NP_V_PP	214	0.195970696	0.065113074
47.1	NP_V_ADVP	13	0.011904762	0.006289545
47.2	NP_V	72	0.489795918	0.089782622
47.2	NP_V_PP	11	0.074829932	0.065113074
47.2	V_NP	7	0.047619048	0.045480173
47.3	NP_V	29	0.420289855	0.089782622
47.3	NP_V_PP	8	0.115942029	0.065113074
47.4	NP_V_NP	11	0.44	0.304779803
47.4	NP_V	9	0.36	0.089782622
47.5.2	NP_V_NP	29	0.420289855	0.304779803
47.5.2	NP_V	16	0.231884058	0.089782622
47.6	NP_V_PP	45	0.323741007	0.065113074
47.6	NP_V	40	0.287769784	0.089782622
47.6	V_PP_NP	4	0.028776978	0.004234793
47.7	NP_V_PP	28	0.184210526	0.065113074
47.7	NP_V_ADVP	10	0.065789474	0.006289545
47.8	NP_V_NP	243	0.399014778	0.304779803
48.1.1	NP_V_PP	427	0.471302428	0.065113074
48.1.1	NP_V	298	0.328918322	0.089782622
48.1.1	NP_V_ADJP	40	0.04415011	0.006790704
48.1.2	NP_V_NP	92	0.51396648	0.304779803
48.2	NP_V	182	0.928571429	0.089782622
48.3	NP_V	240	0.8	0.089782622
48.3	NP_V_PP	26	0.086666667	0.065113074
48.3	S_V	9	0.03	0.011501597

49	NP_V	3	0.1875	0.089782622
50	NP_V_PP	110	0.502283105	0.065113074
50	NP_V	67	0.305936073	0.089782622
50	NP_V_ADVP	13	0.059360731	0.006289545
51.1	NP_V	502	0.331571995	0.089782622
51.1	NP_V_PP	472	0.311756935	0.065113074
51.1	NP_V_ADVP	98	0.064729194	0.006289545
51.2	NP_V_NP	114	0.423791822	0.304779803
51.2	NP_V	46	0.171003717	0.089782622
51.3.1	NP_V	47	0.423423423	0.089782622
51.3.1	NP_V_PP	12	0.108108108	0.065113074
51.3.1	NP_V_NP_PP	10	0.09009009	0.066929775
51.3.2	NP_V	259	0.410459588	0.089782622
51.3.2	NP_V_PP	115	0.182250396	0.065113074
51.3.2	NP_V_S	53	0.083993661	0.08370607
51.3.2	NP_V_ADVP	27	0.042789223	0.006289545
51.4.1	NP_V	7	0.4375	0.089782622
51.4.1	NP_V_PP	3	0.1875	0.065113074
51.4.2	NP_V	103	0.64375	0.089782622
51.6	NP_V_NP	178	0.767241379	0.304779803
51.8	NP_V_NP	65	0.890410959	0.304779803
52	NP_V_NP	118	0.746835443	0.304779803
52	NP_V_S	24	0.151898734	0.08370607
53.1	NP_V_NP	46	0.455445545	0.304779803
54.1	NP_V_NP	222	0.730263158	0.304779803
54.1	NP_V_PP	21	0.069078947	0.065113074
54.2	NP_V_NP	119	0.493775934	0.304779803
54.2	NP_V_NP_NP	35	0.145228216	0.010574453
54.2	S_V_NP	8	0.033195021	0.001240368
54.3	NP_V_NP	101	0.753731343	0.304779803
54.4	NP_V_SBAR	102	0.115384615	0.114514816
54.4	NP_V_NP_PP	63	0.071266968	0.066929775
54.5	NP_V_NP_NP	12	0.069364162	0.010574453
55.1	NP_V	807	0.303497555	0.089782622
55.1	NP_V_S	583	0.219255359	0.08370607
55.1	S_V	381	0.14328695	0.011501597
55.1	NP_V_VP	19	0.007145543	0.002480737
55.1	VP_V	15	0.005641219	0.000325753
55.2	NP_V_NP	137	0.391428571	0.304779803
59	NP_V_NP_S	5	0.009259259	0.003407881
61	NP_V_S	475	0.892857143	0.08370607

62	NP_V_S	745	0.441350711	0.08370607
63	NP_V_NP	45	0.489130435	0.304779803
64	NP_V_S	32	0.432432432	0.08370607
64	NP_V_NP_NP	4	0.054054054	0.010574453
65	NP_V_NP	851	0.772232305	0.304779803
67	NP_V_NP	149	0.465625	0.304779803
68	NP_V_NP	81	0.347639485	0.304779803
68	NP_V_NP_PP	57	0.244635193	0.066929775
69	NP_V_PP	4	0.571428571	0.065113074
70	NP_V_PP	32	0.5	0.065113074
70	NP_V_PP_S	15	0.234375	0.00061392
71	NP_V_S	11	0.55	0.08370607
71	NP_V	3	0.15	0.089782622
76	NP_V_NP	289	0.515151515	0.304779803
76	NP_V_NP_PP	72	0.128342246	0.066929775
77	NP_V_NP	36	0.43373494	0.304779803
78	NP_V_SBAR	96	0.518918919	0.114514816
78	NP_V_NP	59	0.318918919	0.304779803
79	NP_V_NP_PP	15	0.483870968	0.066929775
81	NP_V_NP_PP	57	0.7125	0.066929775
84	NP_V_NP	11	0.366666667	0.304779803
84	NP_V_SBAR	7	0.233333333	0.114514816
85	NP_V_NP	134	0.59030837	0.304779803
85	NP_V_NP_PP	31	0.136563877	0.066929775
9.1	NP_V_NP_PP	308	0.386934673	0.066929775
9.1	NP_V_NP_ADVP	26	0.032663317	0.002004636
9.1	NP_V_PP_NP	12	0.015075377	0.001328071
9.1	NP_V_ADVP_NP	10	0.012562814	0.000651507
9.1	NP_V_PRT_NP	9	0.011306533	0.001064963
9.1	NP_V_PP	60	0.158311346	0.065113074
9.1	NP_V_NP_PP	38	0.100263852	0.066929775
9.3	NP_V_NP_PP	16	0.156862745	0.066929775
9.4	NP_V_NP	319	0.620622568	0.304779803
9.4	NP_V_NP_PP	62	0.120622568	0.066929775
9.4	NP_V_NP_ADVP	3	0.005836576	0.002004636
9.5	NP_V_NP_PP	3	0.3	0.066929775
9.7	NP_V_NP_PP	53	0.25	0.066929775
9.7	NP_V_PP	33	0.155660377	0.065113074
9.8	NP_V_NP	205	0.555555556	0.304779803
9.8	NP_V_NP_PP	26	0.070460705	0.066929775
9.9	NP_V_NP	40	0.4	0.304779803

90	NP_V_NP	216	0.724832215	0.304779803
90	NP_V_NP_PP	28	0.093959732	0.066929775
93	NP_V_NP	39	0.866666667	0.304779803