Distribution Agreement

Laura Manor April 16, 2014

Dictionary Dissection:

A Computational Approach to Recovering Primitive Concepts

by

Laura Manor

Phillip Wolff
Adviser

Department of Linguistics

Phillip Wolff

Adviser

Eugene Agichtein

Committee Member

Marjorie Pak

Committee Member

2014

Dictionary Dissection:

A Computational Approach to Recovering Primitive Concepts

By

Laura Manor

Phillip Wolf

Adviser

An abstract of
a thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Arts with Honors

Department of Linguistics

Laura Manor

Abstract

Dictionary Dissection:

A Computational Approach to Recovering Primitive Concepts

By Laura Manor

There are many linguists who believe that there are universal conceptual primitives that make up the internal structure of all words. I believe that all verbs must contain at least one of these primitives. Although we cannot know what these primitives are, I suggest that some verbs are closer to the conceptual primitives than others. These verbs are the most basic or generic verbs of a language.

Through the computation analysis of a dictionary, I attempt to uncover the most basic of English Verbs. This process includes the iterative re-representation of verb definitions, wherein the verbs in a definition are replaced with the definition of the verbs, and so on. The process, which I refer to as drilling, ends when the re-representation process uses a verb that was used previously.

After completing the process with two different dictionaries, I acquired a list of verbs that share many similarities with theoretical lists put together by linguists in the past. I believe that these basic verbs act as windows to the universal human concepts.

Dictionary Dissection:

A Computational Approach to Recovering Primitive Concepts

By

Laura Manor

Phillip Wolff

Adviser

A thesis submitted to the Faculty of Emory College of Arts and Sciences
of Emory University in partial fulfillment
of the requirements of the degree of
Bachelor of Arts with Honors

Department of Linguistics

2014

Acknowledgements

I would first like to that Dr. Philip Wolff for his guidance in this project and others throughout the year. You create a wonderful environment of learning, creativity, discovery, and camaraderie in the lab. Without your help, none of this would have been possible.

I would first like to thank the members of the Cognitive and Linguistic Systems Lab who provided assistance and encouragement throughout this process. In addition to thanking Lee, Jason, and Hadar for the moral support, I would especially like to thank Kristie who took significant amounts of time to help me verbalize my thoughts, Austin who always came through in a moment of crisis.

I would also like to thank my readers, Dr. Marjorie Pak and Dr. Eugene Agichtein for agreeing to be on the committee for my Honors Thesis Defense as well their patience and understanding.

Finally, I would like to thank Dr. Donald Tuten: without your support, I would have not embarked on this journey.

Table of Contents

Introduction

> I say it would be impossible to define every word. For in order to define a word it is necessary to use other words designating the idea that we want to connect to the word being defined. And if we then wished to define the words used to explain that word, we would need still others, and so on to infinity. Consequently, we necessarily have to stop at primitive terms which are undefined. *(Arnaud and Nicole (1996)/[1662]:64, as cited in Wierzbicka (2002))*

In the 17th century, there were many philosophers who embraced the idea that humans have access to a finite set of universal concepts from which all thought and language is created. Leibniz was one of the first to introduce the idea of a set of basic, innate and universal concepts, which he referred to as "the alphabet of human thoughts." His belief that even the most complex of thoughts was based on these conceptual building blocks was shared by many philosophers of the day, including Descartes, Locke, Pascal, and Arnauld. At the time, analyses like the one suggested by Arnauld and Nicole (1996/[1662]) were unfeasible due to the sheer volume of work involved in the process.

The idea of primitives is still very relevant to current linguistic theory. Most accounts of syntactic and semantic theory rely on the idea that there is a relatively small set of concepts that make up the ideas expressed by language. The biggest problem is that there is no consensus as to what the primitives are. The typical strategy is to compare words together, intuitively looking for common threads either within a language or between languages. This does not work because using intuition is limited due to the sheer number of words we must compare to really grasp an understanding of what primitives might be. This project examines the possibility that these issues can be overcome using computational techniques. In particular, it examines the possibility that we can derive primitives through the process of definitional re-representation.

In effect, I will be using words to describe other words. There are limitations to this idea. In particular, there is good reason to believe that primitives are not directly coded in language due to the variety of ways similar ideas are expressed cross-linguistically. However, I believe that the methods I propose can point us in the direction of primitive words by finding the most semantically basic words in a language. In the extreme, there may be not be a small set of words which are able to describe all other words, in which case the process I suggest would never reach convergence, or the number of words found at the end of the process would be so large that it would not be helpful. However, if there are primitives and they can be roughly captured by a few semantically light words, then the process of definitional re-representation should converge on a relatively small number of verbs. This hypothesis is tested in the current study.

Literary Background

In the following sections, I will discuss various literature that is pertinent to the discussion on basic verbs and primitives. I will begin with a discussion on verbs, including the lexical category, the creation of verb. I will then discuss theories on the internal structure of a word as well as theories of universal primitives and light verbs. I will end with a review of recent semantic classification of verbs.

What Is a Verb?

Before going into detail about the semantic and lexical aspects of a verb, one must first define what a verb is. If you pick up any dictionary and look up the definition of a verb, you will most likely find something along the lines of “a word that characteristically is the grammatical center of a predicate and expresses an act, occurrence, or mode of being,” (Merriam-Webster).

Through inflection, verbs in many languages can encode tense, aspect, mood, voice, and negation as well as information about arguments, such as person, gender, and number.

There are many theories pertaining to lexical categorization, most of which describe two or four major lexical categories. Though these theories vary, they almost always include the distinction between a verb and a noun (Baker, 2003; Haspelmath, 2001). As discussed in Haspelmath (2001), most theories complement these semantically rich categories (content words or lexical categories) from other parts of speech (function words or functional categories).

Though most can agree on what is and is not a verb, different theories suggest various fundamental aspects. While ancient models, like Dionisus Thrax's Tékhnē Grammatiké, focused more on language-specific features such as the marking of tense or mode, modern models attempt to find more universal approaches that are able to describe verbs consistently across all languages. Minimalist theories, which often distinguish between lexical categories using ±V and ±N binary values, assert that verbs are distinguished by the need of a specifier, as discussed in Baker's (2003). Focusing on another aspect, Haspelmath (2001) claims that the most salient feature of verbs is the fact that they appear as predicates without any additional coding, as shown by the following example:

(1a) Plato defined beauty

(1b) *Plato definition beauty

(1c) Plato**'s** definition **of** beauty

(Haspelmath 2001, p. 16541)

Example (1) shows how as a verb, *defined* can easily take the predicate position under *Plato*, but if the verb is replaced with the noun form *definition*, this phrase is no longer grammatical. In English, the extra morphemes *'s* and *of* must be added in order for *definition* to be used as a predicate in this circumstance.

It is not surprising that the most frequent and clearest lexical distinction is between the noun and the verb, as these two categories have the most salient differences cross-linguistically. Even in languages where nouns and verbs can appear in many of the same predicate structures (such as some language families in North America and some Polynesian languages), closer examination reveals other prominent features that distinguish between these main categories (Haspelmath 2001).

Though there are salient distinctions between different parts of speech in English, it is important to note the fluidity that exists as well. Different verbs can encode completely different information. When attempting to analyze a verb, it is important to recognize which features are inherent of a verb, and which features can be expressed through other means. Although there are multiple theories on lexical categorization, what most theories have in common in the fact that a verb is inherently a syntactic category, not semantic. This notion is extremely pertinent when examining theories on lexical decomposition.

Verbal Systems and Word Formation

The English verbal system is a productive open class of words. Although English already has thousands of codified verbs, new ones continue to be created. An informative source that charts the progress of English word formation is the Oxford English Dictionary, which adds new words to their dictionary every three months. In the most recent March 2014 additions, there were 16 brand new verbs (e.g. bookend and hegemonize) along with many other non-verbs added to their 600,000+ word dictionary.

The constant influx of words can be attributed to the English morphology system and the multitude of ways English allows new words to be formed. As discussed in O'Grady (2010), the two most common types of word formation in English are compounding and derivation. Since new English verbs are often based on compounding and derivation, it is commonly believed that these lexical items can be decomposed and paraphrased while still keeping the majority of the semantic information. Other types of word formation include conversion, blending, backformation, clipping, borrowing, and use acronyms.

Even with the extensive lists of ways to form new English words, there are some processes that are rare or impossible in English. Since many theories on lexical semantics focus on word formation, it is important to discuss the ways in which verbs are formed cross linguistically. One such example is incorporation, specifically noun incorporation. Many of the polysynthetic languages of North America rely heavily on noun incorporation to create verbs using various verb roots.

In direct contrast to English's rather flexible verbal system, there are some languages that have relatively small closed classes of verbs. Common in Australia and New Guinea, these languages have a fixed number of verbs ranging from a half dozen to about 250 (Schultze-Berndt 2000, Pawely 2004). As discussed in Schultze-Berndt (2000), the Jaminjung language of Northern Australia features a closed class of about 30 root verbs that combine with an open class of coverbs to create complex predicates. Though this language may seem very radical upon first examination, research by Schultze-Berndt shows how these complex predicates are very similar to the English light verb construction. This idea will be discussed further in a subsequent section.

The Internal Structure of a Verb

The work of the majority of lexical semantics, as well as the present study, is founded upon the assumption that words can be decomposed or paraphrased. There are, however, those who do not believe this is possible. Furthermore, even those who believe this is a valid process disagree on the nature of decomposition.

Theories with no or minimal internal structure.

There are several very distinct theories that suggest a lack of internal verb structure. Championed by Lyons (1968), the first of the theories discussed takes a holistic approach to semantics. Lyons argues that a word in isolation has no meaning, but only acquires meaning through the word's relation to other words. For example, the word *dog* means nothing by itself, but dog can be described as *not a cat*. The meaning of a word is always in contrast to other word and absolute synonymy is nearly IMPOSSIBLE. In this way, a word cannot have an internal structure, since words themselves do not possess meaning.

Another theory that does not allow for internal verb structure is frequently associated with Fodor and heavily influenced by modularity of mind. An extreme nativist, Fodor adamantly refutes any theory that suggests even the slightest variance between individual cognitive structures; according to Fodor, everything must be innate. As discussed in Jackendoff (1990) and Pullman (2005), Fodor claims that all 'natural' mono-morphemic words (i.e. animal, wind) are innate atomic concepts and therefore unanalyzable. As for more unnatural or manmade things (i.e. doorknob, unicorn), he believes that humans have the innate ability to rapidly form concepts for these words, but they too are atomic in nature and are also unanalyzable. Fodor believes that there is no need to create associations between different concepts. For example, the concepts *dog* and *animal* may seem share properties to those who know both English words, but either concept can be understood independently of the other; there is no validity in assuming that one animal encapsulates dog. Fodor cites the lack of satisfactory definitions and of full synonymy as sufficient evidence against any kind of deeper structure.

Theories with complex internal structure.

In 1972, Lakoff published a groundbreaking work on generative semantics that has inspired countless theories, some of which I will discuss later in this section. In his work, Lakoff fashions logical forms using generative grammar syntax as a base for natural logic and abstract predicates such as CAUSE, ALLOW, DIE, or HIT. For example, the word *kill* would take the following form:

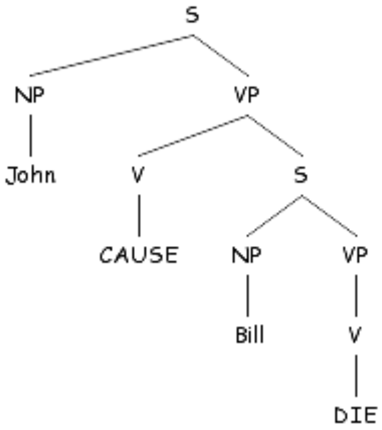

Figure 1. Reprinted from Pullman, 2006

Though this type of analysis, Lakoff gives evidence to the validity of lexical decomposition, and in doing so, suggests the possibility of a finite and universal set of atomic predicates that take sentential complements. While Lakoff does not create a complete list of such predicates, he does discuss the possibility of certain concepts like the predicates mentioned above being on the list.

In a similar vein of thought, Jackendoff (1990) suggests a set of conceptual constituents that map onto an NP (Thing, Event, Property), a PP (Place, Path, sometimes Property) or an S

(State, Event). Using the work of Lakoff (1972) as a starting point, Jackendoff (1990) describes a class of abstract conceptual functions such as GO, STAY, TO, FROM and CAUSE. Together, these constituents and functions create the internal structure of a verb.

While most theories on the nature of meaning in the lexicon focus on semantics, Hale and Keyser (1993) suggest that the basic structure of language is primarily based on abstract verbal syntax at the lexical level. Though they place syntax above semantics, their theory is primarily based on a Lakoff/Jackenfdoff view of lexical semantics, thus their theory supports many of the same claims on lexical decomposition as Lakoff and Jackendoff.

On Universal Primitives and Light Verbs

The idea of a basic or primitive verb has been approached from many different directions over the years. Taking inspiration from 17th century philosophers, linguists such as Goddard and Wierzbicka have attempted to discover semantic primitives through a process called 'reductive paraphrase,' as discussed in Wierzbicka (1972, 2006). Wierzbicka (1972) published a list of a total of 14 hypothetical semantic primitives, which she soon expanded. According to the most recent publications by Goddard and Wierzbica (2002), there are a total of 62 universal human concepts that have been shown to exist in every language they have considered. While each concept has the ability to be expressed through any part of speech, all of the verbs in English reduce to the following 17 semantic primitives: KNOW, THINK, WANT, FEEL, SEE, HEAR, SAY, DO, HAPPEN, MOVE, TOUCH, BE (SOMEWHERE), THERE IS, HAVE, BE (SOMEONE/SOMETHING), LIVE and DIE.

Other theories on primitives include the following theories. Hopper (1991) suggests that the primitive verbs are the most frequently used verbs. Clark (1978) suggests the first verbs that children learn are the most basic. Fodor, as discussed in Jackendoff (1990), views all ‘natural’ mono-morphemic words as primitive.

Others believe that ‘light verbs’ are the most primitive verbs (Butt 2002). Coined by Otto Jeperson in *A Modern English Grammar on Historical Principles* (1909-1949), the term *light verb* was used to describe the V + NP construction in English as a way to add additional descriptive information (e.g. to have a delightful bath). Although the exact meaning might vary from linguist to linguist, current theory suggests that light verbs have very little semantic value and are often used in conjunction with other words to create a more syntactically and semantically complex phrase. This complex phrase expands upon Jeperson’s original model, and is referred to as the light verb construction.

The light verb construction is especially important in languages with small, finite sets of verbs such as the aboriginal Australian languages. By taking these verbs and combining them with additional semantic information to create complex predicates, the finite of verbs can describe infinite situations (Schultze-Berndt 2000). Though this phenomenon may seem exotic, the creation of similar complex predicates is very common throughout the world. Hopper (1992, 1996) even suggests that in vernacular English, this type of complex verbal constructions is preferred to a singular verb.

Attempts of the Semantic Classification of English Verbs

Many theories on primitives suggest looking for primitives in the semantic classification of verbs. In this section I will discuss various published examples of the semantic classifications of English verbs. These databases categorize English verbs (or words) using various syntactic and semantic approaches. Other than the Levin (1993) classes, all of the databases I discuss are readily available online in word-lookup fashion making them extremely accessible.

The Levin Verb Classes. In 1993, Levin published *English Verb Classes and Alternations: A Preliminary Investigation* in which she classified over 3,000 English verbs based on both semantic and syntactic features. The book defines 48 unique classes, each with an average of five subclasses. Each verb falls into at least one of eight syntactic categories as well as at least one of 40 semantic categories. Within the semantic categories, the classes are generally groups of near synonyms, which are then split further into subcategories based upon syntactic features. The following is a list of the different categories in which the verb *crack falls:*

01.1.2.1	Causative/Inchoative Alternation
01.3	Conative Alternation
02.3.4	Swarm Alternation
02.8	"With/Against" Alternation
02.12	Body-Part Possessor Ascension Alternation
07.6.1	Unintentional Interpretation with Reflexive Object
07.6.2	Unintentional Interpretation with Body-Part Object
07.8	Directional Phrases with Nondirected Motion Verbs
43.2	"bang" verbs
45.1	"break" verbs

Though the publication goes quite in depth on each of the categories, Levin warns that the book is "a preliminary large-scale investigation," (p. 17) and "by no means a definitive and exhaustive classification of the verb inventory of English" (p. 18). Although the many categories are interesting in term of finding primitives, I find the fact that the categories are so even unnerving.

WordNet®. Another extensive attempt to map the semantic and lexical features of verbs comes from Princeton University's WordNet. WordNet groups sets of words together to form “sets of cognitive synonyms” called synsets, each of which expresses a unique concept. Each synset is linked to the other synsets via hierarchical “conceptual relations.” There are over 13,000 such verb synsets in the current version. Though I appreciate the attempt to organize words by the sematnic relations, I find the fact that the number of synsets is almost the same as the number of verbs to be rather troubling. Below is an example of the listing for the verb *crack*. (Princeton University, 2010)

<verb.change>S: (v) crack (crack%2:30:01::), check (check%2:30:03::), break (break%2:30:15::) (become fractured; break or crack on the surface only)

<verb.perception>S: (v) crack (crack%2:39:01::) (make a very sharp explosive sound)

<verb.perception>S: (v) snap (snap%2:39:00::), crack (crack%2:39:00::) (make a sharp sound)

<verb.contact>S: (v) crack (crack%2:35:00::) (hit forcefully; deal a hard blow, making a cracking noise)

<verb.change>S: (v) break through (break_through%2:30:02::), crack (crack%2:30:09::) (pass through (a barrier))

<verb.change>S: (v) crack (crack%2:30:02::) (break partially but keep its integrity)

<verb.change>S: (v) snap (snap%2:30:00::), crack (crack%2:30:00::) (break suddenly and abruptly, as under tension)

<verb.social>S: (v) crack (crack%2:41:00::) (gain unauthorized access computers with malicious intentions)

<verb.emotion>S: (v) crack up (crack_up%2:37:00::), crack (crack%2:37:00::), crock up (crock_up%2:37:00::), break up (break_up%2:37:04::), collapse (collapse%2:37:00::) (suffer a nervous breakdown)

<verb.communication>S: (v) crack (crack%2:32:00::) (tell spontaneously)

<verb.change>S: (v) crack (crack%2:30:05::) (cause to become cracked)

<verb.change>S: (v) crack (crack%2:30:07::) (reduce (petroleum) to a simpler compound by cracking)

<verb.change>S: (v) crack (crack%2:30:08::) (break into simpler molecules by means of heat)

VerbNet. VerbNet is the largest on-line lexicon of English verbs (Kipper-Schuler, 2006). VerbNet extends Levin's (1993) verb classes with heavy influence from Korhonen and Briscoe's (2004) propositions regarding the slight reorganizations of classes (Kipper et al, 2006). One of the more unique features of VerbNet is the use of other online lexical resources such as the aforementioned WordNet. VerbNet doesn't just incorporate these resources, but provides links between them, combining the prominent features of each source. VerbNet currently consists of 272 first-level classes.The VerbNet representation of the verb *crack* is "break-45.1, bump-18.4-1, sound_emission-43.2, spank-18.3, (PropBank), (fn Cause_harm), (Grouping)."

Inspiration for the Present Study

I have up until this point discussed various theories on verb and primitives. The remainder of this thesis will assume the following to be true. There exists a finite set of universal concepts, named semantic primitives, that all thought and therefore language is based upon. Humans are able to use the primitives a building blocks that form larger concepts that we represent with arbitrary words; all content words must contain at least one semantic primitive but the majority of content words are the acumination of many primitives. While our language does not allow us to directly access these concepts, there are words which contain very few primitives, known henceforth as basic or generic words. These words can be combined with other content words to describe words which contain more primitives and are therefore denser.

The Present Study

This study attempts to examine the idea of primitive concepts through the recovery of a small group of English verbs which can describe all English verbs. Though there is plenty of theory on the idea of primitives and there have been multiple attempts create sets of primitive concepts, these theories rely on intuition. This does not work because using intuition is limited due to the sheer number of words we must compare to really grasp an understanding of what primitives might be.

In order to uncover these verbs, I propose a computational angle to the process described by Arnaud and Nicole ((1996)/[1662], as cited in Wierzbicka (2002)). This process is the iterative re-representation of verb definitions, wherein the verbs in a definition are replaced with the definition of the verbs, and so on. The process ends when the re-representation process uses a verb that was used previously. In the following sections, I discuss the general procedures, as well as problems faced and the particulars of the sources used.

Obtaining the Definitions

As I have previously mentioned, the idea for this general process is not new, and suggestions for this type of analysis can be dated back to the 17th century. There are two major differences with my project: The first is the fact that I am using an automated process in order to analyze large amounts of date. The second is that I am using a dictionary as my corpus.

At first, using a dictionary might seem almost like cheating, but there are many challenges to overcome. This includes the inherent biases in a dictionary as well as the fact that words have multiple senses.

The Inherent Bias and Inconsistency of Dictionaries. There are inherent biases in the choosing any corpus for analysis. Each corpus is created for a target audience and for specific purposes, which influences how the corpus is written. This must be considered when choosing which dictionary to use.

Though it may seem that dictionaries should be relatively consistent, this is not the case. On the larger scale, each dictionary is written for different audiences. This means that both the verbs defined and the type words used in the definitions vary from dictionary to dictionary. See Appendix A for a comparison of the most frequent verbs used in various corpuses.

On a smaller scale, the majority of dictionaries are written not by an individual, but by multiple people. This could imply that there is less bias in the dictionary since there were multiple people creating the dictionary. On the other hand, this could lead to inconsistency between entries. In an automated program such as the one I created for this thesis, consistency across definitions is necessary. Although an editor may approve all entries in order to improve consistency, this again increases the bias.

Selecting the Dictionaries. The first step of process began with the selection of the dictionaries. During the conceptualization process of this project, Google dictionary was used with great promise. However, Google does not offer an official application programming interface (API) or any other way of easily obtaining the definitions.

After failing to obtain the dictionary used by Google, I went through a similar testing process with multiple other online dictionaries. Of these dictionaries, the Merriam-Webster (M-W) dictionary seemed to be the next best option to Google. M-W provides an easily accessible, official API, created to encourage the use of their dictionary by application developers. M-W offers many dictionaries via API, including Merriam-Webster's Collegiate® Dictionary with

Audio, Merriam-Webster's School Dictionary with Audio (Grades 9-11), and Merriam-Webster's Learner's Dictionary with Audio among others. I chose the Collegiate® Dictionary as it had more than twice as many entries as the other dictionaries.

The process involved in acquiring the M-W definitions was quite somewhat tricky. The API provided by Merriam-Webster exports their definitions in an XML format, with all entries related to the query included. This mean I had to identify the correct entry, and then parse the entry for different senses using string manipulations. See Appendix D for an example API output and for code used to parse the output.

The second dictionary I chose was WordNet's dictionary. As previously mentioned, WordNet is known for its Synset features. In addition, WordNet boasts a dictionary with over 150,000 unique strings, including over 11,000 unique verbs. The process for obtaining the definitions from WordNet was much simpler than from M-W as WordNet is seamlessly integrated with Python through the Natural Language Tool Kit (NLTK).

Preparing the Definitions. Once the definitions were obtained, it was necessary to prepare the definitions for the drilling. In order to identify the verbs, I ran each of the definitions through the Stanford Factored English Parser and marked each verb identified using parentheses. However, as definitions of verbs are not complete sentences, it was difficult for the parser to pick up many of the verbs. This was especially true of verbs that appear often as nouns, such as *question* and *chance*. To fix this, for each dictionary, I printed a list of definitions that did not identify verbs as well as definitions that did not recognize a verb as one of the two first words, and manually added identification markers to the verbs in these definitions.

In order to facilitate the movement of information between different steps of the process, I used JSON (JavaScript Object Notation). JSON is a language independent text format that allow for the seamless transfer of information between different programming languages. Created by ECMA International, data from Java, Python, Pearl, C, C++, and many other languages are written in a format that is read easily both humans and computers.

Drilling

After the definitions were in the proper format, I could then begin the process of re-representing the verbs in the definitions, which I refer to as *drilling*. My algorithm closely mirrored the process described by Arnauld, which replaced the verbs in a definition with the definition of that verb, one step at a time. Below is a slightly modified example of the decomposition of the verb *crack* , which eventually falls into a *cause/make* cycle.

Iteration	Chain	Definition
1	crack	(**break**) apart
2	crack → break	((**separate**) into parts with suddenness or violence)
3	crack → break → separate	(((**set**) apart) into parts with suddenness or violence)
4	crack → break → separate → set	((((**cause**) to (**sit**)) apart) into parts with suddenness or violence)
5	crack → break → separate → set → cause	(((((**make**) something (happen) or (exist)) to (sit)) apart) into parts with suddenness or violence)
6	crack → break → separate → set → cause → make	((((((**cause**) to (happen)) something (happen) or (exist)) to (sit)) apart) into parts with suddenness or violence)

As can be seen in the example above, many of the definitions had multiple verbs. Thus, I decided to create two similar but slightly different processes: one that drilled every verb in the dictionary, and one that focused on only the first verb. When drilling all verbs, I stopped after the seventh iteration. When drilling just the first verb I stopped after each definition identified a chain. Merriam-Webster took eleven iterations, while WordNet took eight iterations to reach this point. To see the code used to complete this process, see Appendix E.

Multiple Senses. Part of the need for dictionaries in the first place is that words are often difficult to describe. Over time, new words are created, and others change meaning or function. It is often hard to describe the meaning of a word in a broad enough manner when at the same time explaining the specifics of that verb. This leads to the use of multiple senses for many entries in the dictionary. Some of these senses are more frequent than others, some more specific, and some are downright archaic. This leads to difficulty when trying to analyze definitions using computational methods.

In the process I am using for this analysis, it is necessary to replace a word with its definition. But what happens when there are multiple senses? Which one do you choose? It would have been impractical (and not entirely automated) to go through every definition and find the best-suited definitions for each specific instance. For this reason, I decided to use the first sense listed in the respective dictionary. Though for my project it would be best to use either the most frequent or the most general definition, I was limited by the choice of the editor.

Focus on the First Verb in the Definition. When beginning this process, I had simply been attempting to drill every verb in the definition. However, it soon became clear that not only did this method take a significant amount of time, but the results were not much better than if we were to take a simpler approach and only consider the first verb in the definition. This decision to focus on the first verb was not made lightly and was supported by various theories regarding lexical decomposition, general semantics, and light verbs in particular.

As previously discussed, theories of lexical decomposition suggest that a word can be broken down into different parts. Theory on light verbs suggests that most semantically complex verbs can be broken down into complex predicates containing more semantically light verbs. Generally, when verbs are broken down, they tend to be a lighter verb with additional information attached. Some of this information is expressed with other verbs, and some with other parts of speech. However, this first verb seems to contain the most important information regarding the original idea, with the latter information complementing the main verb. By focusing on just the first verb of a definition, we may be losing some semantic information given by a second or third verb in the definition, but I consider this information to be expressed when looking at the chain through the previous verb in the chain.

Initial Drilling problems with Merriam-Webster. Unfortunately, after obtaining the results from M-W, I noticed some of the entries were not as expected. I realized something was strange when the traditionally semantically light verbs such as *cause* and *make* were dropping significantly in the frequency lists. This was cause for concern, so I compared the definitions I found online with the definitions that I obtained through the API and found some discrepancies.

One of the main reasons for choosing M-W was because the definitions that appeared first in their online dictionary were optimal for this project as these were very descriptive, yet generalized enough to be used in a variety of contexts. These definitions were often not included in the Collegiate API.

After this discovery, I decided to manually edit the 25 most frequent verbs. For each of these verbs, I found the official M-W definition and, if applicable, I inserted the definition found online into the primary sense in my dictionary. No senses were deleted, but rather the existing senses were re-ordered.

Initial Drilling Problems with WordNet. Though acquiring definitions through WordNet was simple, there were some major downfalls. As previously discussed, WordNet was created for the semantic organization and categorization of words. The dictionary function, while convenient, is a secondary feature of WordNet. Consequently, many of the definitions were sub-optimal. First, many of the verbs were defined as the verb itself. For example, the first definition listed for the definition of "reach.v.01" was "reach a destination, either real or abstract." Though this may be useful in other situations, this created a problem for my algorithm. The second issue was that WordNet did not seem to order their senses in a way that had the most basic definition first, but often had very narrow entries listed as the first definition.

Just as I manually edited the top 25 most frequent verbs in Merriam-Webster, I identified the definitions that were causing problems and edited them to better suit the algorithm. For verbs in which the primary sense was defined using the verb itself, I switched the sense with a more general definition listed as a later sense that did not contain the key verb. In this way, the previous primary sense would still be represented, and when the verb was found in the definition of another verb, it would not simply be replaced.

Results and Discussion

This study was heavily based on discovery. Rather than pick a set of primitives and attempt to re-create them, I wanted to recover these primitives through analysis of the dictionaries. The first step, then, is to identify these primitives. As previously discussed, in the algorithm created, verbs will eventually begin to repeat themselves and form cycles. I theorized that these cycles represent the most basic of English verbs. Using the cycles that can be found in Appendix B, the verbs in Table 1 were recovered by the drilling process.

Table 1 shows the verbs that remained after drilling was complete. The first column shows the verbs found in the cycles, the second column (‘inst’) is the number of senses that the verb appears in as the first cyclic verb, and the third column is the percentage of total senses the verb appears in as the first cyclic verb. There were some glitches in the program which created a single instance of a repeated word, these verbs are shown in the OTHER category.

M-W First Verb Only		
verb	#	%
go	4655	22%
make	4140	20%
cause	3305	16%
move	3212	15%
put	2674	13%
be	1244	6%
have	421	2%
place	327	2%
use	265	1%
get	259	1%
hold	152	1%
gain	111	1%
establish	77	0%
pay	76	0%
OTHER	42	0%
acquire	35	0%
choose	25	0%
select	24	0%
recognize	19	0%
acknowledge	16	0%
confront	7	0%
face	6	0%
institute	4	0%
throb	3	0%

WN First Verb Only		
verb	#	%
cause	4554	23%
move	3927	20%
do	3658	18%
carry	2169	11%
have	2144	11%
put	1555	8%
keep	577	3%
change	510	3%
place	184	1%
fail	131	1%
hold	128	1%
reach	119	1%
engage	87	0%
stay	52	0%
ask	30	0%
OTHER	35	0%
pick	29	0%
live	26	0%
protect	23	0%
select	17	0%
inquire	12	0%
censure	12	0%
remain	12	0%
evaluate	10	0%

Table 1. Post-Drilling Frequencies for Merriam-Webster and WordNet

The variation between the verbs found in Merriam-Webster and WordNet shows the bias presented by each dictionary (see Table 1). One of the largest differences between the two was that M-W found mostly binary cycles whereas over 80% of WordNet verbs converged into the six-verb-long *cause/do/engage/carry/move/change* chain. However, the six most commonly found generics in Merriam-Webster account for a whopping 92% of the verbs. Even if this data is not perfect, it indicates that there is validity in the theory that all English verbs can be represented by a small number of basic verbs.

Though the verbs found by Merriam-Webster and WordNet are slightly different, what is most astonishing about the lists of verbs recovered is the similarity between these verbs and the those discussed by previous theory on primitives such as verbs Lakoff (1972) suggested, events suggested by Jackendoff (1990) or concepts Wierzbecka (2002) suggested. Compared to Wierzbecka (2002) the main group of concepts missing from the verbs found revolves around senses (WANT, FEEL, SEE, HEAR, SAY) and life (LIVE, DIE). Neither Jackendoff nor Lakoff produced extensive lists of the concepts they use in their theories, however all the concepts commonly associated with their ideas can be found on the list except for the ones that are also missing from Wierzbecka's list.

Conclusion and Suggestions for Further Research

As one can see from the results of this study, dictionaries are not as straightforward as they might appear. Though some mild success was observed in terms of uncovering basic English verbs, the polysemous nature of English verbs seemed to hinder the results. While some dictionaries may be set up for this type of analysis, the definitions used for this thesis seemed to be sub-optimal. For example, verbs often appeared within their own definitions in WordNet when the word could easily be deconstructed. After making some edits to the definitions, there were some trends that began to appear that closely followed theories on primitive concepts.

As discussed, there is an inherent bias in dictionaries that reflects in both sublet and overt differences between dictionaries. Though they are not written overtly for this type of analysis, dictionaries are still written by linguists who may carry the bias of previous linguistic theory with them when chosing which words to use in the definitions. The range and variation between dictionaries may prove to be an asset when looking at a larger amount of data, however I only looked at two dictionaries. I would not hesitate to suggest completing the same process with many more dictionaries to see if the same trends occur. By somehow combining the data received from multiple dictionaries, we might be able to reduce the inherent bias of a single source.

Another way to improve the data would be to introduce a better way of identifying not only the primary sense of a verb, but also which sense to use when drilling. Ideally, when replacing a verb with its definition, the program would be able to select the best sense for the context. However, this seems to be quite a challenge. I may be possible to create a stronger algorithm that identifies some basic syntactic information such as the number of arguments in order to select a more optimal sense.

A final way to really improve the data would to find a better way to extend the scope of this type of analysis by using not just verbs, but all non-functional parts of speech, such as nouns and adjectives. This project would require a much larger investigation considering the number of nouns in English.

Though the results from the present thesis were not conclusive, there is evidence that all English verbs do converge to a relatively small set of English verbs that are able to describe all other. With a larger corpus, an added syntactic component and the systematic analysis of all lexical items, it may yet be possible to realize a more cohesive lists of the most basic words in English.

References

Baker, M. C. (2003). Lexical categories: Verbs, nouns and adjectives. Cambridge University Press.

Bird, S, Loper, E. & Klein, E. (2009), Natural Language Processing with Python. O'Reilly Media Inc.

Butt, M. (2003, June). The light verb jungle. In Workshop on Multi-Verb Constructions (pp. 1-28).

Clark, J. L. D. (1978). Direct testing of speaking proficiency. Princeton, NJ: Educational Testing Service.

Collegiate® Dictionary (2014). Merriam Webster, Incorperated.

Ecma International (2013). The JSON Data Interchange Format.

Fellbaum , C. (1998, ed.) WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.

Goddard, C. (2010) Universals and variation in the lexicon of mental state concepts. In Malt, B. & Wolff P. (Eds.) Words and the Mind: How words capture human experience. New York, NY: Oxford University Press, pp

Goddard, C. & Wierzbica, A. (Eds.) (2002). Meaning and universal grammar Volumes I&II. Philadelphia, Pa : John Benjamins Pub. Co.

Hale, K. & Keyser, S. J. (1993). On argument structure and the lexical representation of semantic relations. In The view from Building 20, Cambridge, MA, MIT Press.

Haspelmath, M. (2001). Word classes and parts of speech. In N. J. Smelser & B. Baltes (eds.), International Encyclopedia of the Social and Behavioral Sciences. 24--16538.

Hopper, P. J. (1991). Functional Explanations in Linguistics and the Origins of Language. Language and Communication, 11, 45-47.

Jackendoff, R. (1979). Toward an Explanitory Semantic Representation. Linguistic Inquiry, 7(1), 89-150.

Jackendoff, R. (1990). Semantic structures. Cambridge, MA: MIT Press.

Jespersen, O. (1942). A modern English grammar on historical principles. Part iv, Morphology. Ejnar Munksgaard.

Kemp, A. The Techne grammatiké of Dionysius Thrax. Translated into English.

Keyser, S., & Hale, K. (1986). Some Transitivity Alternations in English (Vol. 7, pp. 381-416). Lexicon Project Working Papers.

Kipper, K., Korhonen, A., Ryant, N., & Palmer, M. (2006, June). Extending VerbNet with novel verb classes. In Proceedings of LREC Vol. 2006, No. 2.2, p. 1)

Klein, D. & Manning, C. D. (2003). Fast Exact Inference with a Factored Model for Natural Language Parsing. In Advances in Neural Information Processing Systems 15 (NIPS 2002), Cambridge, MA: MIT Press, pp. 3-10.

Korhonen, A., & Briscoe, T. (2004, May). Extended lexical-semantic classification of English verbs. In Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics (pp. 38-45). Association for Computational Linguistics.

Lakoff, G. (1972). Linguistics and Natural Logic. In D. Davidson & G. Harmon (Eds.), Semantics of Natural Language. D. Reidel Publishing.

Levin (1993) English Verb Classes and Alternations: A Preliminary Investigation

Merriam-Webster Online Dictionary (2012). Merriam-Webster, Incorporated.

O'Grady, W. (2010). Contemporary linguistics: An introduction. Boston, MA: Bedford/St. Martins.

Oxford English Dictionary. (1989). Oxford: Oxford University Press.

Pawley, A. (2006). Where have all the verbs gone? Remarks on the organisation of languages with small, closed verb classes. Rice University Linguistics Symposium, 16-18 March 2006. http://www.ruf.rice.edu/~lingsymp/Pawley_paper.pdf.

Princeton University (2010). About WordNet. http://wordnet.princeton.edu.

Pulman, S. G. (2005). Lexical decomposition: For and against. In Charting a New Course: Natural Language Processing and Information Retrieval, pp. 155-173. Springer Netherlands.

Schuler, K. (2007). VerbNet: extensions and mappings to other lexical resources. http://www.coll.unisaarland.de/projects/salsa/workshop/contents/workshop _ slides/slides4.pdf.

Schultze-Berndt, E. (2000). Simple and complex verbs in Jaminjung: A study of event categorisation in an Australian language. Nijmegen: Dissertation, Univ. Nijmegen (MPI Series in Psycholinguistics 14).

Wierzbicka, A. (1972). Semantic primitives. Frankfurt am Main : Athenäum,

Wierzbicka, A. (2006). Semantic Primitives. In Brown, K (Ed) Encyclopedia of Language & Linguistics (Second Edition). Elsevier, Oxford, pp. 134-137.

Appendix A

Verb Freqency Before Drilling

	MWSENSEALL			MWSENSEFIRST			WNSENSEFIRST			WNSENSEALL			COMBINED**			WIKIPEDIA		
	verb	inst	%*	verb	inst	%	verb	inst	%	verb	inst	%	verb	inst	%	verb	inst	%
1	make	1564	4.6%	make	1367	6.5%	make	1426	7.1%	make	1759	5.5%	make	6116	5.7%	be	6941786	14.8%
2	be	1020	3.0%	cause	769	3.6%	be	856	4.3%	be	1679	5.2%	be	4072	3.8%	have	1277132	2.7%
3	use	970	2.9%	be	517	2.5%	cause	699	3.5%	cause	892	2.8%	cause	3220	3.0%	become	589309	1.3%
4	cause	860	2.5%	become	501	2.4%	move	435	2.2%	move	599	1.9%	become	2054	1.9%	make	522288	1.1%
5	become	618	1.8%	give	494	2.3%	become	387	1.9%	become	548	1.7%	move	2007	1.9%	include	498810	1.1%
6	give	595	1.8%	move	408	1.9%	give	387	1.9%	give	509	1.6%	give	1985	1.9%	take	482773	1.0%
7	move	565	0.6%	take	355	0.7%	take	346	0.7%	take	500	0.7%	take	1697	0.7%	use	414775	0.5%
8	take	496	0.6%	put	333	0.7%	put	316	0.6%	have	419	0.7%	use	1491	0.6%	play	402486	0.5%
9	bring	409	0.5%	bring	328	0.7%	have	278	0.6%	put	380	0.6%	put	1411	0.6%	go	342970	0.5%
10	have	385	0.5%	have	302	0.7%	remove	270	0.6%	remove	323	0.5%	have	1384	0.6%	win	342121	0.5%
11	put	382	0.5%	use	227	0.6%	come	208	0.5%	come	287	0.5%	bring	1130	0.5%	come	278529	0.4%
12	come	294	0.5%	come	205	0.6%	provide	181	0.5%	get	260	0.5%	come	994	0.5%	begin	256092	0.4%
13	go	289	0.5%	form	197	0.5%	get	178	0.5%	go	244	0.5%	remove	971	0.5%	leave	249514	0.4%
14	form	279	0.5%	go	191	0.5%	bring	167	0.5%	bring	226	0.5%	go	876	0.5%	serve	240092	0.4%
15	pass	242	0.5%	remove	167	0.5%	go	152	0.4%	form	216	0.5%	form	835	0.5%	give	240052	0.4%
16	set	225	0.5%	set	159	0.5%	form	143	0.4%	provide	213	0.4%	get	728	0.5%	work	236869	0.4%
17	remove	211	0.6%	place	155	0.7%	place	134	0.7%	do	212	0.7%	provide	719	0.7%	get	233403	0.5%
18	produce	203	0.6%	provide	146	0.7%	cover	129	0.6%	use	209	0.7%	place	616	0.6%	do	229204	0.5%
19	provide	179	0.5%	pass	145	0.7%	cut	116	0.6%	cut	181	0.6%	pass	615	0.6%	find	227098	0.5%
20	place	177	0.5%	produce	145	0.7%	change	115	0.6%	cover	172	0.5%	set	590	0.6%	lead	211301	0.5%
21	hold	175	0.5%	get	130	0.6%	act	107	0.5%	act	171	0.5%	cover	585	0.5%	appear	196731	0.4%
22	cover	174	0.5%	draw	117	0.6%	turn	102	0.5%	change	168	0.5%	cut	578	0.5%	receive	196408	0.4%
23	cut	169	0.5%	cut	112	0.5%	express	100	0.5%	turn	151	0.5%	act	549	0.5%	move	195005	0.4%
24	draw	163	0.5%	utter	111	0.5%	utter	92	0.5%	place	150	0.5%	produce	522	0.5%	provide	189500	0.4%
25	act	162	0.5%	cover	110	0.5%	pass	88	0.4%	hold	145	0.5%	change	518	0.5%	return	187191	0.4%
26	get	160	0.5%	strike	110	0.5%	use	85	0.4%	keep	141	0.4%	hold	509	0.5%	call	186053	0.4%
27	strike	147	0.4%	act	109	0.5%	hold	82	0.4%	pass	140	0.4%	turn	473	0.4%	follow	184933	0.4%
28	express	146	0.4%	hold	107	0.5%	perform	82	0.4%	express	136	0.4%	express	464	0.4%	continue	184744	0.4%
29	perform	141	0.4%	subject	103	0.5%	set	81	0.4%	perform	134	0.4%	do	442	0.4%	die	183687	0.4%
30	turn	139	0.4%	engage	102	0.5%	hit	79	0.4%	leave	130	0.4%	perform	441	0.4%	say	182701	0.4%
31	change	136	0.4%	change	99	0.5%	keep	79	0.4%	lose	125	0.4%	draw	440	0.4%	know	181387	0.4%
32	keep	131	0.4%	furnish	97	0.5%	travel	79	0.4%	set	125	0.4%	keep	429	0.4%	start	180555	0.4%
33	engage	127	0.4%	treat	95	0.5%	treat	78	0.4%	write	112	0.3%	utter	416	0.4%	run	180078	0.4%
34	subject	117	0.3%	reduce	86	0.4%	fill	75	0.4%	break	106	0.3%	strike	406	0.4%	remain	173148	0.4%
35	furnish	116	0.3%	perform	84	0.4%	leave	74	0.4%	fall	106	0.3%	engage	388	0.4%	see	171105	0.4%
36	fall	116	0.3%	express	82	0.4%	do	73	0.4%	produce	106	0.3%	treat	370	0.3%	write	162223	0.3%
37	write	115	0.3%	undergo	82	0.4%	walk	73	0.4%	travel	106	0.3%	undergo	358	0.3%	hold	161526	0.3%
38	utter	113	0.3%	turn	81	0.4%	show	71	0.4%	show	102	0.3%	reduce	338	0.3%	live	150413	0.3%
39	treat	111	0.3%	keep	78	0.4%	arrange	70	0.4%	undergo	101	0.3%	break	333	0.3%	allow	147331	0.3%
40	do	111	0.3%	mark	78	0.4%	undergo	69	0.3%	add	100	0.3%	hit	326	0.3%	join	146905	0.3%
41	reduce	110	0.3%	serve	73	0.3%	break	68	0.3%	stop	100	0.3%	show	323	0.3%	help	143501	0.3%
42	specify	108	0.3%	throw	70	0.3%	look	68	0.3%	utter	100	0.3%	fall	322	0.3%	lose	140885	0.3%
43	carry	107	0.3%	drive	68	0.3%	produce	68	0.3%	draw	99	0.3%	force	322	0.3%	show	136852	0.3%
44	undergo	106	0.3%	deprive	67	0.3%	add	67	0.3%	force	99	0.3%	mark	316	0.3%	create	133907	0.3%
45	throw	105	0.3%	apply	66	0.3%	mark	66	0.3%	engage	97	0.3%	write	316	0.3%	feature	129279	0.3%
46	force	104	0.3%	force	66	0.3%	create	65	0.3%	hit	96	0.3%	work	314	0.3%	produce	127334	0.3%
47	serve	102	0.3%	increase	66	0.3%	lose	65	0.3%	look	95	0.3%	fill	313	0.3%	end	124814	0.3%
48	drive	102	0.3%	carry	65	0.3%	strike	64	0.3%	work	94	0.3%	leave	310	0.3%	build	124758	0.3%
49	work	101	0.3%	hit	64	0.3%	work	64	0.3%	run	92	0.3%	travel	309	0.3%	tell	120383	0.3%
50	obtain	101	0.3%	send	64	0.3%	play	63	0.3%	fill	90	0.3%	throw	307	0.3%	try	120201	0.3%
	verbs	2495		verbs	1816		verbs	1573		verbs	2145		verbs	2916		verbs	N/A	
	instance	33934		instance	21099		instance	20000		instance	32080		instance	107113		instance	46778964	

Appendix B

Merriam Webster

cycle	appearances	
go/move	7867	37.3%
cause/make	7519	35.6%
place/put	3209	15.2%
have/hold	1811	8.6%
acquire/get/gain	405	1.9%
use?	57	0.3%
establish/institute	81	0.4%
choose/select	49	0.2%
acknowledge/recognize	35	0.2%
face/confront	13	0.1%
be?	6	0.0%
throb/pulsate	4	0.0%
pay?	2	0.0%
procreate/beget	3	0.0%
other	38	0.2%

WordNet

cycle	appearances	
cause/do/engage /carry/move/change	16644	83.2%
have/posses	2146	10.7%
keep/hold/retain	714	3.6%
fail	131	0.7%
reach	119	0.6%
remain/stay	64	0.3%
pick/select	46	0.2%
ask/enquire	42	0.2%
live/inhabit	30	0.2%
protect/shield	26	0.1%
evaluate	10	0.1%
censure/rebuke	15	0.1%
sweep	6	0.0%
like/prefer	4	0.0%

Table 3. Post-Drilling Cycles for Merriam-Webster and WordNet

Appendix C

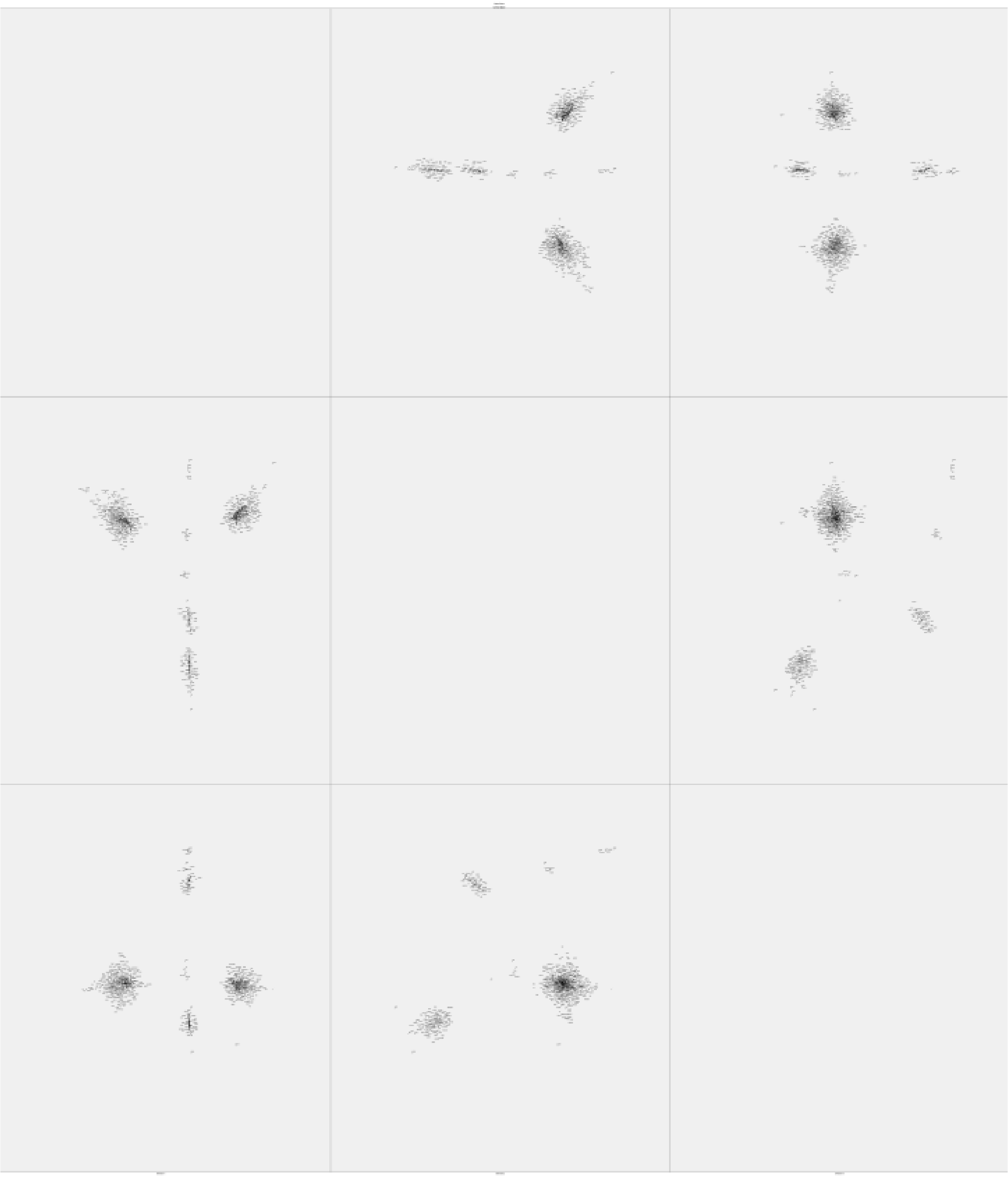

Figure 2. Spatial Representation of English Verbs in Relation to Basic Verbs

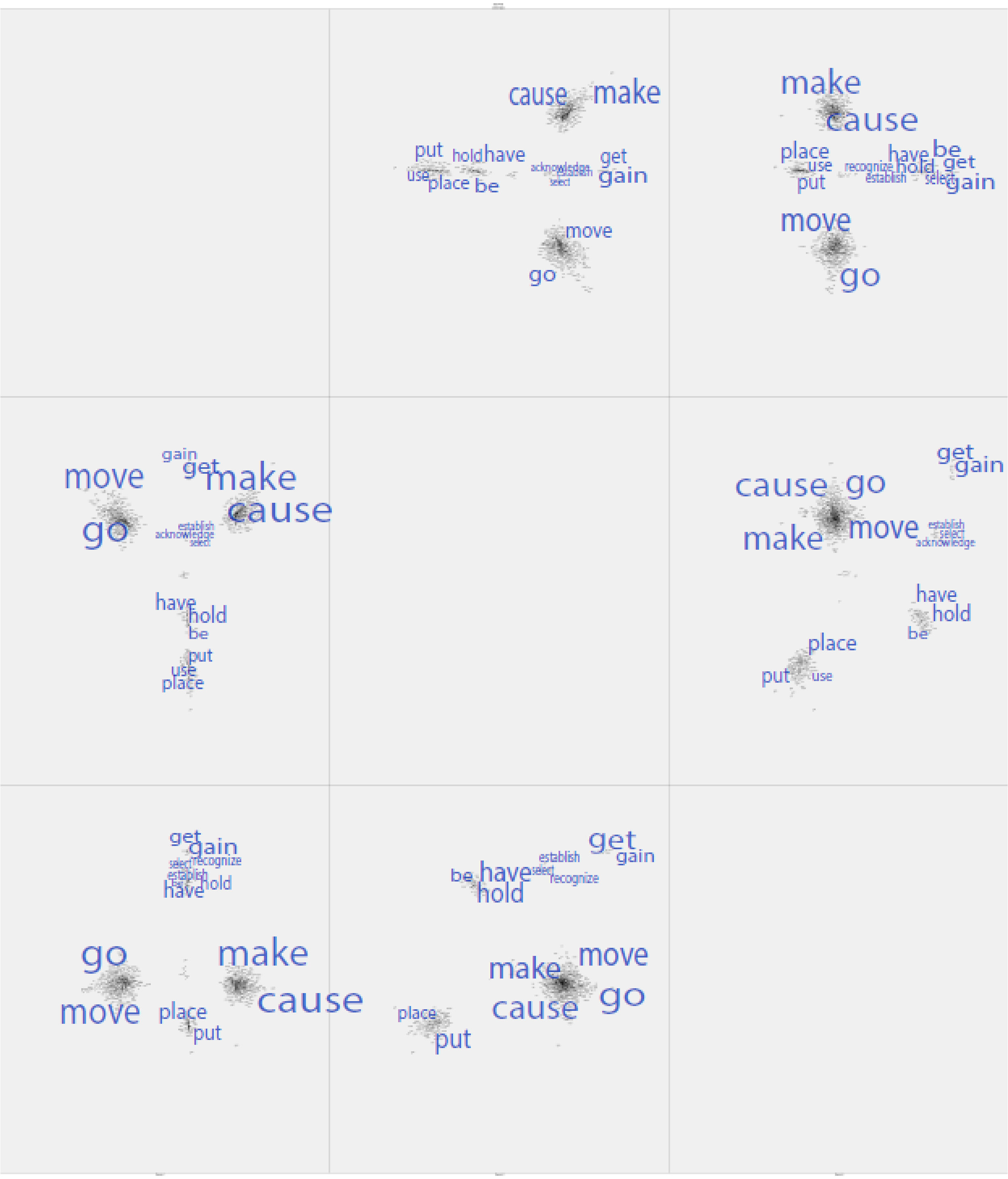

Figure 3. Spatial Representation of English Verbs in Relation to Basic Verbs with Labels

Appendix D

The following is an example of output from Merriam-Webster Collegiate Dictionary API for the key *crack*.

<fl>verb</fl><et>Middle English <it>crakken,</it> from Old English <it>cracian;</it> akin to Old High German <it>chrahhōn</it> to resound</et><def><vt>intransitive verb</vt><date>before 12th century</date> <sn>1</sn> <dt>:to make a very sharp explosive sound </dt> <sn>2</sn> <dt>:to break, split, or snap apart</dt> <sn>3</sn> <dt> :<sx>fail</sx>: as</dt> <sn>a</sn> <dt>:to lose control or effectiveness under pressure <un>often used with <it>up</it></un></dt> <sn>b</sn> <dt>:to fail in tone </dt> <sn>4</sn> <dt>:to go or travel at good speed <un>usually used with <it>on</it> </un></dt><vt>transitive verb</vt> <sn>1 a</sn> <dt>:to break so that fissures appear on the surface </dt> <sn>b</sn> <dt>:to break with a sudden sharp sound </dt> <sn>2</sn> <dt>:to tell especially suddenly or strikingly </dt> <sn>3</sn> <dt>:to strike with a sharp noise :<sx>rap</sx> </dt> <sn>4 a <snp>(1)</snp></sn> <dt>:to open (as a bottle) for drinking</dt> <sn><snp>(2)</snp></sn> <dt>:to open (a book) for studying</dt> <sn>b</sn> <dt>:to puzzle out and expose, solve, or reveal the mystery of </dt> <sn>c</sn> <dt>:to break into </dt> <sn>d</sn> <dt>:to open slightly </dt> <sn>e</sn> <dt>:to break through (as a barrier) so as to gain acceptance or recognition</dt> <sn>f</sn> <dt>:to show or begin showing (a smile) especially reluctantly or uncharacteristically</dt> <sn>5 a</sn> <dt>:to impair seriously or irreparably :<sx>wreck</sx> </dt> <sn>b</sn> <dt>:to destroy the tone of (a voice)</dt> <sn>c</sn> <dt>:<sx>disorder</sx> <sx>craze</sx></dt> <sn>d</sn> <dt>:to interrupt sharply or abruptly </dt> <sn>6</sn> <dt>:to cause to make a sharp noise </dt> <sn>7 a <snp>(1)</snp></sn> <dt>:to subject (hydrocarbons) to <fw>cracking</fw></dt> <sn><snp>(2)</snp></sn> <dt>:to produce by cracking </dt> <sn>b</sn> <dt>:to break up (chemical compounds) into simpler compounds by means of heat</dt></def><dro><drp>crack the whip</drp> <def><dt>:to adopt or apply an authoritative, tyrannical, or threatening approach or policy (as in demanding harder work from employees)</dt></def></dro><dro><drp>crack wise</drp> <def><dt>:to make a wisecrack</dt></def></dro>

The following is the Python module used to obtain definitions from M-W.

```
def addDefinition(verb):
    try:

        url="http://www.dictionaryapi.com/api/v1/references/collegiate/xml/"
                +verb+"?key=080e99c0-959b-4d57-9805-a47ec38dc1c7"
    except:
        print "!!!!!!there was and issue with " + verb

    obj=urllib.urlopen(url);
    content = obj.read()
    obj.close()

    while "<vi>" in content:
        start = content.find('<vi>')
        stop = content.find('</vi>')
        content = content[:start]+content[stop+5:]

    while True:
        if content.find("<entry id=\"" + verb) < 0:
            print verb + " was not found in m-w dictionary"
            return
        content = content[content.find("<entry id=\"" + verb):]


        if content.find("<fl>verb</fl>") < 0 :
            print verb + " does not have a verb entry"
            return

        firstHalf = content[:content.find("<fl>verb</fl>")]
        secondHalf = content[content.find("<fl>verb</fl>"):]

        content = secondHalf[:secondHalf.find("</entry>")]
        print content
        break
    index = 1


    while "<dt>" in content:
        start = content.find("<dt>")
        end = content.find("</dt>")

        sense = content[start:end]
        content = content[end+5:]
        while "<" in sense:
            start = sense.find("<")
            stop = sense.find(">") + 1
            sense = sense[:start]+sense[stop:]
        sense = sense[1:]
        newKey = verb + "[" + str(index) + "]"
        global newDict
        newDict[newKey] = sense
        index += 1
```

Appendix E

```
def iterateFirst(iter, origDict, newDict, start, stop):


    if start is 0:
        freqSenseDict, freqSenseList =
                        Lists.getSenseFrequencyListFirst(newDict)
        writeFile(freqSenseDict,"freqSenseDict" + iter+ "0.txt")
        writeFile(freqSenseList,"freqSenseList" + iter+ "0.txt")

        freqEntryDict, freqEntryList =
                        Lists.getEntryFrequencyListFirst(newDict)
        writeFile(freqEntryDict,"freqEntryDict" + iter+ "0.txt")
        writeFile(freqEntryList,"freqEntryList" + iter+ "0.txt")
        start+=1



    for x in range(start,stop):
        iter = "_firstverb_wn3_" + str(x)
        print "iteration " + iter
        newDict = Drill.drillFirst(origDict,newDict)
        writeFile(newDict,"iterations"+iter+".txt")

        freqSenseDict, freqSenseList =
                        Lists.getSenseFrequencyListFirst(newDict)
        writeFile(freqSenseDict,"freqSenseDict"+iter+".txt")
        writeFile(freqSenseList,"freqSenseList"+iter+".txt")

        freqEntryDict, freqEntryList =
                        Lists.getEntryFrequencyListFirst(newDict)
        writeFile(freqEntryDict,"freqEntryDict"+iter+".txt")
        writeFile(freqEntryList,"freqEntryList"+iter+".txt")
        print "after iteration " + iter + " there are " +
                  str(len(freqSenseList)) + "/" +
                  str(len(freqEntryList)) + " verbs"

def drillFirst(startDict, oldDict):

    print "starting the process"
    number = float(len(startDict))
    done = 0.00000
    percent = 0

    tempDict = dict()
    for key in oldDict:
        verbKey = key[:key.find(".")]
        #find (verbs)... replace with XVERBX and add to list
        newDef = oldDict[key]

        verbs = []
        while newDef.find("(") >= 0:
            l = newDef.find("(")
            r = newDef.find(")")

            verb = newDef[l+1:r]
            base = getBase(verb)
            for x in range(1,10):
                if x is 9:
                    newDef = newDef[:l]+newDef[l+1:r]+ newDef[r+1:]
                    tempDict[key] = newDef
                    print verb + "/" + base + " not in dict"
                    break
                if (base + ".v.0" + str(x)) not in startDict.keys():

                    print verb + "/" + base + str(x) + " not in dict"
                    continue


                newDef = newDef[:l] + "[" +  startDict[base+".v.0"+str(x)]
                                        + "]"+ newDef[r:]
                tempDict[key] = newDef
                break

            break


        done+=1
        p = int(done/number*100)
        if p>percent:
            if p%10 is 0:
                percent = p
                print str(p) + "% done drilling"

    return tempDict


def getBase( verb):
    base = wn.morphy(verb, wn.VERB)
    return str(base)
```