The Thesis committee for Juan Daniel Pinto
Certifies that this is the approved version of the following thesis:

Creating a Conversational Hebrew Vocabulary List

APPROVED BY
SUPERVISING COMMITTEE:

Esther L. Raizen, Supervisor

Elaine K. Horwitz, Co-Supervisor

Creating a Conversational Hebrew Vocabulary List

by

Juan Daniel Pinto

Thesis

Presented to the Faculty of the Graduate School
of the University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

Master of Arts

The University of Texas at Austin
May 2018

Dedication

Interdum et malesuada fames ac ante ipsum primis in faucibus. Aliquam congue fermentum ante, semper porta nisl consectetur ut. Duis ornare sit amet dui ac faucibus. Phasellus ullamcorper leo vitae arcu ultricies cursus. Duis tristique lacus eget metus bibendum, at dapibus ante malesuada. In dictum nulla nec porta varius. Fusce et elit eget sapien fringilla maximus in sit amet dui.

Acknowledgements

Interdum et malesuada fames ac ante ipsum primis in faucibus. Aliquam congue fermentum ante, semper porta nisl consectetur ut. Duis ornare sit amet dui ac faucibus. Phasellus ullamcorper leo vitae arcu ultricies cursus. Duis tristique lacus eget metus bibendum, at dapibus ante malesuada. In dictum nulla nec porta varius. Fusce et elit eget sapien fringilla maximus in sit amet dui.

Mauris eget blandit nisi, faucibus imperdiet odio. Suspendisse blandit dolor sed tellus venenatis, venenatis fringilla turpis pretium. Donec pharetra arcu vitae euismod tincidunt. Morbi ut turpis volutpat, ultrices felis non, finibus justo. Proin convallis accumsan sem ac vulputate. Sed rhoncus ipsum eu urna placerat, sed rhoncus erat facilisis. Praesent vitae vestibulum dui. Proin interdum tellus ac velit varius, sed finibus turpis placerat.

Creating a Conversational Hebrew Vocabulary List

by

Juan Daniel Pinto, M.A.
The University of Texas at Austin, 2018
SUPERVISORS: Esther L. Raizen, Elaine K. Horwitz

Interdum et malesuada fames ac ante ipsum primis in faucibus. Aliquam congue fermentum ante, semper porta nisl consectetur ut. Duis ornare sit amet dui ac faucibus. Phasellus ullamcorper leo vitae arcu ultricies cursus. Duis tristique lacus eget metus bibendum, at dapibus ante malesuada. In dictum nulla nec porta varius. Fusce et elit eget sapien fringilla maximus in sit amet dui.

Mauris eget blandit nisi, faucibus imperdiet odio. Suspendisse blandit dolor sed tellus venenatis, venenatis fringilla turpis pretium. Donec pharetra arcu vitae euismod tincidunt. Morbi ut turpis volutpat, ultrices felis non, finibus justo. Proin convallis accumsan sem ac vulputate. Sed rhoncus ipsum eu urna placerat, sed rhoncus erat facilisis. Praesent vitae vestibulum dui. Proin interdum tellus ac velit varius, sed finibus turpis placerat.

Table of Contents

Dedication

Acknowledgements **v**

Abstract **vi**

Contents **vii**

List of Tables **ix**

1 Introduction **1**

2 Background: Review of the literature **4**

2.1 Corpus design 4

2.1.1 Corpus size 5

2.1.2 Text types 10

2.2 List design 14

2.2.1 General use vs. specialized use 15

2.2.2 Identifying words (word family levels) 16

2.2.3 Objective design 20

3 Methods: Creating the Conversational Hebrew Vocabulary List (CHVL) **24**

3.1 The corpus 25

3.2 Cleansing the corpus 28

3.3 Reading data 31

3.4 Calculations 33

3.4.1 Frequency 34

3.4.2 U_{DP} (dispersion) 34

3.5 Sort and export 37

4 The CHVL: A vocabulary list of conversational Modern Hebrew **39**

4.1 Challenges and future direction 41

4.1.1 Methodological challenges 41

4.1.2 Functional challenges 46

5 Implications for other less commonly taught languages **51**
5.1 Easy reproducibility and growth 51

Appendix A: Conversational Hebrew Vocabulary List (CHVL) **52**

Appendix B: Scripts **84**
Appendix B.1: HebrewLemmaCount.py 84
Appendix B.2: OMDb-fetch.py 92
Appendix B.3: single_file_extract.py 94

Appendix C: Movies used **95**

References **96**

List of Tables

1 Sample of the first 30 items on the CHVL. 39
2 Breakdown of coverage percentages. 41

1 Introduction

This thesis provides an in-depth look at the creation of the Conversational Hebrew Vocabulary List (hereafter CHVL)—a list of the most common words in spoken Modern Hebrew. Its two-fold aim is (1) to explore the theory behind the creation of the CHVL, along with implications for similar projects, and (2) to describe the methods and provide the tools to make the process as reproducible as possible.

The complete list itself, consisting of 5,000 items, is included as an electronic supplement and can be downloaded free of charge.[1] A partial list of the first 1,000 items can be found in *Appendix A*.

A review of the literature will first highlight the gap that exists for less commonly taught languages (LCTLs). Because the overwhelming majority of the previous research in vocabulary frequency lists has focused on English (and a handful of other European languages), some important nuances are yet to be addressed. More often than not, the few non-English word lists that do exist, along with much of the research in vocabulary acquisition, have taken at face value some of the findings of this limited-scope research—often without questioning whether the same methodologies and conclusions should be applied to different languages.

The present paper is, therefore, an effort to partially fill that gap in order to help educators interested in creating and/or using word lists for their own classrooms, for wider dissemination, or simply for general research purposes. In doing so, it will provide an overview of some of the key decisions that must be taken into account for such a project.

The various uses of word frequency lists can be loosely classified into research applications and practical applications. Examples of research applications include traditional linguistic studies that look for common morphological patterns, corpus-linguistic studies seeking to understand language through "real world" texts, and psycholinguistic studies that explore connections between a speaker's mental lexicon and word frequency. Practical applications of word lists include curriculum and

[1] Supplements can be downloaded directly from the thesis archive of the University of Texas at Austin. A separate repository at GitHub also contains the complete CHVL at *https://github.com/juandpinto/opus-lemmas*.

textbook planning for language teachers, vocabulary selection for graded readers and dictionaries, and even independent language study. Of course, some of the most influential studies straddle both sides of this divide and attempt to answer questions such as: How can vocabulary knowledge be appropriately tested and measured? What is the role of extensive reading (as opposed to intensive reading) in incidental vocabulary acquisition? What level of vocabulary do learners need in order to read extensively for pleasure? What level of vocabulary do learners need in order to succeed in an academic setting? What role does specialized vocabulary play in reaching understanding? These questions and their answers rely heavily on the creation and use of trustworthy word frequency lists. Yet due to the resources and effort required to create these lists, they are rarely found for less commonly taught languages.

The primary research question guiding this project is this: *What are the most-frequently used words in conversational Modern Hebrew?* The resulting study also addresses the following secondary research questions, which were necessary to address in order to answer the aforementioned question: *What effect does a corpus of unvocalized texts have on the identification of word families in the computerized creation of a vocabulary frequency list? What factors affect the way that boundaries are demarcated for various levels of word families in Modern Hebrew?* And finally: *What implications might these findings have for word list creation and use as it pertains to other less commonly taught languages?*

The literature review will serve as a basis for many of the important decisions taken during the creation of the CHVL. These decisions—surrounding both corpus and list creation—along with their reasoning, will be explained further in an analysis of the literature. For the sake of clarity, these decisions are listed here at the outset. They are as follows:

Corpus design - *Size:* - *Text types:* The corpus consists of a single text type: conversation. This is to best fit with the list's intended audience. In order to accomplish this, movie and television subtitles compose the core of the corpus. **List design:** - *Use:* The primary intended audience for the CHVL is composed of beginning-to-low-intermediate learners of Hebrew as a foreign language. It is designed for both receptive and productive language use. - *Word family levels:* The word family level that is best suited for the CHVL's intended audience is the lemma. - *Criteria:* The

CHVL was created using exclusively objective criteria, meaning that it is the product of calculations, and it was not manually tweaked in any way. The empirical criteria used were frequency and range.

Following the review of literature and explanation of theory, the process of the CHVL's creation will be explained in detail, along with findings from the project. Possible implications for other less commonly taught languages will then be discussed. Finally, the CHVL and any scripts used will be provided in the appendices.

2 Background: Review of the literature

The theoretical foundation for the creation and use of word frequency lists rests on the observation, made popular by the linguist George Kingsley Zipf in the 1930s and 40s, that if one were to create a frequency list of words in a large enough text, the first word would occur roughly twice as often as the second word, three times as often as the third word, and so on (1935, 1949).

This exponential distribution is significant because it means that a small number of words make up the bulk of a text, whereas the majority of the words occur very few times(Sorell, 2012). Paul Nation, one of the most influential scholars in the field of vocabulary acquisition, has pointed out that Zipf's Law—as it is has come to be known—can serve as motivation to language learners and teachers, since learning the most common vocabulary in a language covers so much of the communication that naturally occurs (2013, p. 34).

This observation guides the entire endeavor of word list creation and use. Though the CHVL is not sorted using raw frequency alone[2], the effect of Zipf's law can be easily seen in the listed frequencies that accompany each item on the list.

One level above this theoretical basis lie the theoretical considerations of the process that serve as the structure upon which the CHVL is built. These include corpus size and text type, general vs. specialized lists, word family levels, and objective criteria. Each of these issues will be treated separately throughout this literature review.

2.1 Corpus design

Before designing a word list, a careful, clear plan must be made for the design of the corpus from which the list is extracted. The corpus must be representative of the language context that the word list wishes to analyze. Of course, it is impossible to capture all of the communications that take place in a particular language. For this simple reason, researchers must make do with an approximation of the whole: a bounded corpus of language.

[2]The sorting method is explained in the sections *Objective Criteria*, *Dispersion*, and *Sort and Export*.

Though the focus of this literature review is the creation of word frequency lists, the truth is that relatively few corpora have been created for this specific purpose. Most corpora have aimed at being general collections that cover the language (usually English) as a whole in an attempt to serve different theoretical and applied uses. Yet despite this broad objective, the creation of corpora has historically revolved around two big questions: (1) how large should the corpus be, and (2) what kinds of texts should it include. These questions are important not only for corpus creation, but also for corpus selection. Both of these points will be addressed here, with the recurring emphasis being corpus use for word list creation.

2.1.1 Corpus size

Conventional wisdom in corpus creation states that more is better. If a word list is to accurately reflect the frequencies of words in the language as a whole, then a corpus must contain enough text to approximate the overall use of discourse. This line of thinking is equivalent to the maxim in quantitative research that a sample should be as representative of the target population as possible. And in order to maximize the statistical probability of this representation, the sample must be of an appropriate size for the study.

True, larger sample sizes often increase this probability, but they also tend to be more resource-intensive for the researcher. The same is true of corpus size. When creating a vocabulary list, then, what is an "ideal" corpus size?

Corpora composed of millions of tokens are easy to access today. This is especially true of corpora of written material—corpora of spoken language are still comparatively small. And thanks to advances in computing power, it is finally becoming plausible for more researchers without access to extensive resources to use these mega-corpora for the purpose of word list creation.

The first project to create a one-million-token corpus was a joint effort by Henry Kučera and W. Nelson Francis of Brown University to compile a corpus of American English texts printed in 1961 (Kučera & Francis, 1967), known today simply as the *Brown Corpus*. They strived to create a corpus with equal amounts of texts from different sources by randomly selecting 500 passages of 2,000 words each from

different published materials found at the Brown University Library and the Providence Athenaeum. This mixed design would be used as a model by many of the corpora created during the next few decades: . These began to be compiled at increasingly faster rates. Many of these corpora were created—in part—to serve as parallel corpora of different varieties of English.

As an example of how quickly corpora have grown in recent decades, consider the history of COBUILD. What began in 1980 as a collaboration between Collins Publishing and a group of researchers led by John Sinclair—the Collins Birmingham University International Language Database (COBUILD)—led to the creation of the *Collins Corpus* of 7-million-tokens by 1982. It continued expanding until transforming into the *Bank of English* in the 1990s, which reached 320 million words in 1997. In 2005, as part of the Collins World Web, which also comprises French, German, and Spanish corpora, it reached 2.5 billion words (*Collins Cobuild English grammar*, 2005). The Collins Corpus now contains over 4.5 billion words ("The history of Collins COBUILD," n.d.).

Today, with the use of web-crawling applications that scour the internet and collect text at unprecedented speed, the sky's the limit.The *enTenTen12* corpus is composed of 12 billion English tokens, all of which were collected in 12 days (Jakubíček, Kilgarriff, Kovář, Rychlý, & Suchomel, 2013)! At what point, then is a corpus sufficiently large for word list creation?

Studies have approached this specific problem by creating multiple frequency lists—from varying sizes of corpora—and then comparing the efficacy of these lists themselves. The way that efficacy is operationalized, however, varies among studies.

Some studies have explored how closely the rankings of items on a word frequency list correlate with reaction times in a lexical decision task—a widely-used procedure in psychological and psycholinguistic research. In a lexical decision task, participants are presented with a series of words and non-words, one after the other, and they are asked to judge which is which as quickly as possible. The reaction times are then analyzed for each word. It is generally agreed that the average time it takes participants to react to a word is a reflection of the way the mental lexicon is organized. For our purposes, multiple studies have found that there exists an inverse correlation between word frequency and reaction time on a lexical decision task (Balota &

Chumbley (1984); Whitney (1998)). In other words, more common words are accessed and recognized more quickly than less common words. Therefore, an effective word frequency list should correspond to and reflect this reality.

This was precisely the approach taken by Brysbaert & New (2009), who compared respond times collected as part of the massive Elexicon Project (Balota et al. (2007)) to words on a series of frequency lists made from increasingly larger corpora. The corpora used were all subcorpora extracted from the British National Corpus (BNC). With each subsequent increase in token count, the word list correlated more and more closely with the response times from lexical decision tasks. This observation validates the line of thinking described at the beginning of this section regarding the need for large corpora. Brysbaert and New hoped to find an "ideal" corpus size after which the increase in effectiveness would no longer be significant enough to justify the additional cost of resources. After conducting several regression analyses on the two sets of data, they found that the variance in the response times that could be accounted for by corpus size reached a plateau at about 16 million words. In other words, for corpora with less than 16 million words, the size of the corpus had a significant effect on the correlation between word frequencies and average response times for those words on lexical decision tasks. For corpora with more than 16 million words, the effect of increasing corpus size became considerably more subtle. In the end, they concluded that in order to construct an effective word list for *high-frequency* words, a corpus of about 1 million tokens is needed. However, in order to reach the same effectiveness for *low-frequency* words, a corpus size of at least 16 million words is preferable.

A different, more straightforward methodology is to directly compare word lists made from corpora of different sizes. Rather than judging the "effectiveness" of a list, this approach measures similarities shared between different lists. Hypothetically, doing this at increasing corpus sizes should allow one to find a size after which the variance between lists only minimally decreases. As with the previous approach, the goal here is to find a point at which the benefits of increasing size no longer outweigh the additional needed resources.

Essentially, then, all corpora of sufficient size should result in nearly the same word frequency list—a theory based on a strict interpretation of Zipf's law applied to all

natural language. If the appropriate criteria can be found—Sorell (2013) suggests—then this would, at last, provide a solution to Nation's (2013, p. 24) observation that, problematically, word lists tend to disagree rather drastically on both the words included and their respective ranking.

Inspired by the computational linguistic measure of *rank distance* (Popescu & Dinu, 2008)—a method for comparing stylistic differences between texts—Sorell developed a variant of this methodology (2013). First, he used different corpora of the same size to create multiple word lists, one for each corpus, ranked entirely by frequency. He then identified the percentage of words that are *not* shared between each set of two lists. Finally, he averaged these percentages to find the level of variability created at that specific corpus size. The levels of variability he found were remarkably close to each other—despite using a wide variety of entirely different corpora (with no overlap on texts within each one). He then increased the size of each corpus and repeated the process.

In order to calculate this level of variability, Sorell used a modified version of a complex formula that he borrowed from the natural sciences, and called his resulting calculation the *Dice distance*. Though this Sørensen–Dice coefficient that he altered (also known by other names) is widely used in botany and other fields[3] to measure similarity in areas and samples of different sizes (Dice, 1945; Sørensen, 1948), the frequency lists measured by Sorell were all purposefully of the same size. What this means is that—apparently without realizing it—his *Dice distance* was ultimately just a simple percentage:

$$\frac{\textbf{number of different words between frequency lists}}{\textbf{total size of frequency list}}$$

Regardless of the round-about way he used to calculate it, Sorell's resulting measure for each corpus size—the level of variability—can be accurately described as the average proportion of difference for word lists at that particular corpus size.

Sorell found that a stable list (about 2% variation) of the most frequent 1,000 words, or a reasonably stable list (less than 5% variation) of the most frequent 3,000, words

[3]It has even been used in corpus linguistics studies before, primarily as a way to measure collocation (Rychlý, 2008).

can be created using a corpus of 50 million tokens. In other words, 1,000-type word lists created from different 50-million-token corpora will likely only differ by 20 words. At the 3,000-type level using the same sizes of corpora, the lists will likely vary by less than 150 words. This is a remarkable level of similarity. Expanding the list to 9,000 types will still only have about 4–7% variation, or 360–630 words. Even corpora of 20 million tokens can be considered sufficient in many cases, since they will result in 3,000-type word list with roughly 5% variation, and 9,000-type word list with less than 10% variation.

Taking a similar approach, though with significant variations, Brezina and Gablasova (2015) compared four corpora of various sizes: The Lancaster-Oslo-Bergen Corpus (LOB), The BE06 Corpus of British English (BE06), The British National Corpus (BNC), and EnTenTen16. These corpora had respective token sizes of 1 million, 1 million, 100 million, and 12 billion. The word list created from each corpus was, in this case, a combination of frequency and dispersion—a measure that will be discussed in more detail later in this paper. The resulting word lists were then compared, and the percentage of shared vocabulary words calculated. Additionally, the researchers also calculated the correlation between the ranking for each word that was shared between word lists. Contrary to Sorell, Brezina and Gablasova considered this final comparison an important part of understanding the effect of corpus size.

The aim of this study was not to find a corpus size after which the difference was negligible, but rather to find if there was a significant difference between word lists made from corpora of different sizes. The study found a 78%–84% overlap between each of the 3,000–lemma word lists. 71% of the words were shared among all four of the lists. Based on this number, Brezina and Gablasova concluded that regardless of corpus size—at least for anything larger than one million tokens—"similar results" are obtained.

This conclusion differs significantly from Sorell's, who concluded that a corpus of at least 20 million tokens (though 50 million is preferable) is needed for a stable word list with low variability. These disagreements are primarily the result of a difference in what should be considered "stable." At 71% vocabulary overlap—which is sufficient for Brezina and Gablasova—870 words were only found in one of the four lists. This is drastically higher than Sorell's threshold, which at the 3,000-word level varies

in roughly 150 words. Note that Nation and Hwang (1995) found a level of overlap similar to Brezina and Gablasova when comparing the GSL, the LOB, and the Brown corpora—a percentage of overlap that they deemed to be not particularly high. As Nation later put it, "Brezina and Gablasova are a bit too tolerant in accepting that 71% or even 78%-84% overlap is good enough. If roughly one out of every four or five words is different from one list to another, that is a lot of difference" (2016, p. 100).

Another difference to mention between these two studies is the unit of counting used. Sorell made lists based on *types*, whereas Brezina and Gablasova preferred the use of *lemmas*. I will explain this important distinction in a later section of this review ("Identifying Words"). For now, it is sufficient to say that the effect of these different measures in comparing word lists created from corpora of different sizes has (to my knowledge) not been studies. This is one area that could benefit from further research.

Lastly, the corpora used by Brezina and Gablasova were all-inclusive: each built on its own philosophy on the way that different types of texts should be balanced in a corpus, but all seeking to be representative of English as a whole. This is also true of the corpora used by Brysbaert and New in their study using response times from a lexical decision task. Contrast this with Sorell's word lists, which were systematically created from corpora that consisted of only one specific text type. Surely, this is a factor to consider in corpus design.

Therefore, having a sufficiently large corpus is important, as demonstrated in this section. But is it enough? How much do the types of texts included in a corpus factor into its effectiveness for word list creation?

2.1.2 Text types

There's been a lot of debate about the "best" way to balance a corpus' text types. This is a major aspect of corpus design, and one worth delving into. At the end of the day, much of it comes down to the purpose of the corpus. When used for the creation of word lists, one must also consider the intended purpose of the word list itself. Is it for general use or for one of many possible specialized uses? More on this

in the next section.

In order to design a corpus with different amounts of text types (i.e. narrative, conversational, academic), clear definitions for these text types are necessary. But is there a better way than the use of subjective genres to classify texts?

Or is there a better methodology than simply mixing a bunch of different texts together, with the hope that the resulting word list covers the language as a whole? This is the most common way of creating frequency lists, but it tends to result in a mix of words that have little relevance to any one purpose. Esoteric, academic words in a beginners' vocabulary list? Science fiction terms in a vocabulary list for business managers? It's obvious that a list is only as good as the corpus from which it's made, which is why a clear delineation of different text types and their qualities is critical.

When speaking of corpus balance, I refer to the proportion of different text types that make up a corpus. Published corpora have taken different approaches in this regard, and published word lists have made use of a variety of strategies for balancing the corpora from which they are made. Coxhead's *Academic Word List* (2000) was created from a carefully-designed corpus that used equally-sized sub-corpora of texts from different disciplines. This suited the purpose of her word list well, since it was intended to serve students from a variety of disciplines.

The importance of identifying a taxonomy of text types based on objective criteria: are there distinguishable linguistic differences between an informal correspondence and a narrative work of fiction? What about between a romance and a fantasy novel?

Biber's early work (1988) conducted an analysis of a wide variety of texts using large corpora to tag syntactic markers and other linguistic attributes that could potentially be used to define different types of texts. In this study, he found a series of five categories (each consisting of two opposite ends of a spectrum) in which texts varied: involved vs. informational, narrative, situated vs. elaborated, persuasive, and abstract. He then conducted a very large study, which he published as a book, (1995) that found eight distinct, recurring patterns of different combinations of these categories. These groupings serve as a linguistically-based taxonomy that divides texts along objective lines, rather than subjective, culturally-defined genres.

Similar but independent studies were conducted for Somali, Korean, Nukulaelae Tuvuluan, Taiwanese, and Spanish (Biber, 1995; Jang, 1998). For each language, a unique set of text types were identified. However, the texts were found to align along similar distinguishing linguistic dimensions as the English texts.

Sorell (2013) sought to simplify Biber's eight text types into categories suitable for corpora study. He did this by noticing the closely similar ways that some of the text types lined up along Biber's five linguistic categories, also incorporating some extra-linguistic features, such as shared contexts (e.g. predominantly spoken types). He also dropped Biber's two smallest text types, deeming them impractical for corpus study and difficult to isolate. In doing this, he came up with four simplified text types: interactive (conversation), general reported exposition (general writing), imaginative narrative (narrative writing), and academic. Regarding this last type, Biber's study found a nosignificant difference between academic writing in the natural sciences ("scientific exposition") and the humanities ("learned exposition")—he found that natural science uses more concrete language, whereas the humanities tend to use more abstract language. However, Sorell sought to unify these for the sake of simplicity, simply leaving their distinction to "a future study" (2013, p. 68). Sorell acknowledged that his wasn't the first attempt at simplification of Biber's text types, a surprisingly similar effort having been made in the *Longman Grammar of Spoken and Written English* (Biber, Johansson, Leech, Conrad, & Finegan, 1999, p. 16) and the *Longman Student Grammar of Spoken and Written English* (Biber, Conrad, & Leech, 2002, p. 23).

Sorell found that each of his four simplified text types yielded a vocabulary frequency list that was as unique as the linguistic criteria that Biber had used. He also measured how different they were from each other, and found all four to be equidistant from the next in this order: conversation, narrative, general writing, and academic writing (See section on corpus size for an explanation of this measurement). Sorell, therefore, claims that his own study of vocabulary frequency using his simplified text types as a base has "validated Biber's studies by adding a vocabulary dimension to the description of each of the key text types" (2013, p. 201).

Despite the importance of spoken language—or the conversation text type—for language learners and linguistic studies, the number of conversation corpora that exist,

as well as their size, is very limited. This is clearly because of the difficulty of gathering large amounts of spoken data that then needs to be transcribed by hand in order to be analyzed. It is true that speech recognition software has come a long way in recent years, but its rate of error remains too high for research purposes. It has been estimated that it takes 40 hours to professionally transcribe one hour of audio recording, making the task too costly. For this reason, some researchers have begun looking at alternative sources for a conversation corpus, including the internet and movie subtitles.

New, et al. (2007) created a 50-million-token corpus of French subtitles. They divided this into four subcorpora, one for each of the type of media from which the subtitles were extracted: French films, English movies, English television series, and non-English-language European films. The reason for using French subtitles from English media is the sheer dominance of English in the film industry. In order to counter-balance the much larger sizes of the two subcorpora extracted from English media, the researchers measured word frequencies for each subcorpora separately, then averaged them to arrive at the final frequency used for their ranked word list.

In order to test the validity of their new approach, New, et al. used two different methods. First, they compared their subtitle word list with word lists created from more traditional corpora. Second, they used lexical decision times—similar to Brysbaert and New (2009) above—to test the rankings of words on their list.

The first test found a .73 correlation with a classical French spoken corpus, the "Corpus de Référence du Français Parlé" (CRFP; Equipe DELIC, 2004). However, when looking at the specific words and semantic categories that differ the most, it's clear that most major differences are caused by the monologue-nature of the CRFP. This corpus was created from a large number of interviews (each asking the same questions to the interviewee), whereas movie subtitles tend to be composed primarily of people interacting in conversations. This results in more colloquial expressions having higher frequencies in the subtitle corpus. The nature of movies themselves also played a role, resulting in an overrepresentation of words related to action movies and police matters—words like *tuer* [to kill], *prison* [jail], and *armes* [weapons] (p. 665).

For the second test of the subtitle word list, the researchers used the lexical decision

times from two previous experiments. They found that the subtitle list's ability to predict lexical decision times was at least equally as accurate as the CRFP frequencies or those from a traditional corpus of written French. In many cases, it actually fared much better, surprising even the researchers themselves. However, this latter test was based on the rather small sample sizes of the two previous experiments (234 and 240 words), limiting the reliability of this test.

Picking up on these findings, and expanding the lexical decision task to a much larger sample size, Brysbaert and New (2009) compiled a corpus of English subtitles ($\text{SUBTLEX}_{\text{US}}$) and evaluated it as part of their study. This corpus is composed of subtitles from a wide variety of American films since 1900, though a majority are from 1990, as well as a large number of American television series. They found that the subtitle frequencies were especially good at predicting the lexical decision times of short words, often surpassing the accuracy of rankings based on the many written corpora they tested. It had more difficulty explaining the response times of longer words, which are more rarely found in film than in literature. Overall, their own conclusion confirmed that of the New, et al. (2007) study, that word frequencies derived from subtitle corpora seem to have a clear advantage over other types of corpora.

Though these two studies arrive at the same conclusion regarding the use of subtitles, more research is needed in this area. If, indeed, subtitles can be considered as appropriate sources for corpora of the conversation text type, their availability will open many possibilities previously made nearly impossible by the difficulty of the collection medium.

2.2 List design

Perhaps even more complex than appropriately designing the corpus from which to extract vocabulary for a word list, researchers have found a wide range of variables that play a role in the design of the list itself. Questions addressed in the literature deal with the difference between a general service list and a specialized list, differences in the way that a "word" is defined and measured, different ranking criteria used, and the influence of subjective criteria on list creation, among other issues.

2.2.1 General use vs. specialized use

Nation (2016) emphasized the importance of identifying the purpose of a word list before beginning the creation process. He believes that the main purpose of most general-use lists is to select vocabulary that language learners should learn during their first years of study. Though this may be the stated goal of some general-use lists, it is clear that they in fact serve a wide variety of purposes. He rightfully suggests, however, that the goal of serving language learners is far too broad to be very helpful. Language learners come to the task at different ages, with different language needs, and with different reasons for learning the language. A word list that is useful for adult learners intent on attending university will likely not be helpful for young leaners whose language focuses on animals, colors, and other age-appropriate material. And yet general-use lists are far more common than specialized-use lists. This is largely due to attempt at finding the language's core vocabulary.

The majority of word lists in use attempt to describe the vocabulary of the language as a whole. They are designed to be broad and all-encompassing so that they can serve any number of uses and scenarios. Essentially, they are lists that are created for general use. This broad nature of general use lists is reflected in the name of the most widely-used word list, West's *General Service List* (1953). Others include Nation's BNC/COCA lists, Browne's *New General Service List* (2014), Brezina and Gablasova's *New General Service List* (2015), and Dang and Webb's *Essential Word List* (Nation, 2016).

Another way of understanding general-use lists is that their objective is to find what is often termed the *core* vocabulary. Though not always explicitly stated, the philosophy behind this approach is that the language being used—usually English—has at its center a self-contained lexicon of essential, primary, basic, fundamental vocabulary that then runs through the entire language. There are layers of frequency and increasing complexity beyond this, with regions of specialized language demarcated for specific purposes such as fields of study or external dialects. Still, this core vocabulary is at the center of it all, and the purpose of a word list is to identify what words fall within its boundaries. Sorell (2013) evaluated a number of definitions of core vocabulary found in the literature. He suggests that general use lists, such as West's GSL, serve as intuitively-selected lists of core written communication, whereas

survival vocabulary lists—often found in travel guides or similar materials—are core vocabulary lists of oral communication.

Relatively fewer researchers have created word lists aimed at a more specific purpose or target audience. Specialized-use lists can be designed to only include words that belong to a specific domain, such as a discipline or trade. They can also encompass vocabulary found in a broad range of disciplines, but which are common in a specific context, such as academic texts. In this case, they usually serve as supplements to aid language learners who are already familiar with the core vocabulary of the language.

Perhaps the most well-known example of a specialized-use list is Coxhead's Academic Word List (2000), which replaced the University Word List (Xue & Nation, 1984) as the go-to vocabulary list for aspiring students intent on attending an English-speaking university or those entering the academic world. This is considered a *general* academic word list, since it is for academic use in general, and not for a specific discipline.

More specialized lists include those designed for business English courses, or medical English courses. This is sometimes designated *technical vocabulary*. Nation (2016) explains that technical vocabulary is most often taught after students have mastered general-use vocabulary, and after they have some familiarity with academic vocabulary. Chung and Nation (2003) looked into the nature of a technical vocabulary. By studying specialized words in the fields of anatomy and applied linguistics, they found that a large number of technical words are also found in the language's core vocabulary, or have a general academic use as well. However, when used in a technical text, these words take on a specialized definition that is particular to that domain. This means that much vocabulary is shared across layers of vocabulary, though they may vary semantically, based on context.

2.2.2 Identifying words (word family levels)

One of the most essential questions that needs to be answered when designing a word list is how one is defining a *word*. Though this may seem like a straightforward decision, it requires thorough planning and a solid understanding of the

theory behind the decision. Should *jump* and *jumped* be counted as two different words or just one? What about irregular inflections such as *go* and *went*? In an article aimed at raising awareness of what he calls the "*Word* dilemma," Gardner points out that the validity of much vocabulary research hinges "on the various ways that researchers have operationalized the construct of *Word* for counting and analysis purposes" (2007, p. 242).

The literature has generally come to accept some key terms that are helpful when speaking of the way words are counted. Beginning with the most basic measurement and progressing to the most complex, we can choose to count tokens, types, lemmas, or word families.

Measuring *tokens* means simply measuring the total number of words. The sentence "I like small dogs, big dogs, and every other kind of dog" contains twelve tokens—twelve words in total. Counting *types* refers to the number of separate and distinct words. That is, *dog* and *dog* are the same type, but *dogs* is a different type—even a single difference makes them different types. The sentence above is composed of eleven types. A level above this, the *lemma* includes the stem of the word and its inflected forms, but not any derived forms of the word (derived forms are usually considered a different part of speech). So *do*, *does*, and *did* are all the same lemma, but *doable* is not. This is because *doable* has the derivational affix *-able*, which turns it into an adjective. Francis, et al. define lemma as "a set of lexical forms having the same stem and belonging to the same major word class, differing only in inflection and/or spelling" (1982, p. 1).

Finally, the term *word family* is used to describe an even more inclusive level than the lemma. However, its precise definition has often varied among researchers. Bauer and Nation (1993) sought to rectify this problem through an in-depth classification of English affixes. Borrowing from Thorndike's (1941) study of English suffixes, their grouping was based on a series of eight criteria: frequency, productivity, predictability, regularity of the written form of the base, regularity of the spoken form of the base, regularity of the spelling of the affix, regularity of the spoken form of the affix, and regularity of function (1993, pp. 255–256). They identified seven "levels" of word families, with each successive one including a larger number of affixes, and therefore a larger number of types per word family. One very useful aspect of their

particular system is that it places all the previous levels (type, lemma, etc.) within the same framework. Under their schema, a level 1 word family is the same as a type, a level 2 word family is a lemma (including all regular inflected affixes), and level 7 (the highest level) consists of classical roots and affixes beyond what most speakers any longer consider separate affixes.

Nation himself suggests that for the purposes of language learning, these specific family word levels can be used simply "as a starting point as an initial framework of reference" (2016, p. 36). That is, they are one interpretation of how to systematically count words for a frequency list. These levels are based on criteria that reflect the needs of language learners, rather than on any psycholinguistic theory of how speakers' mental lexicon is arranged. Still, the idea of word families aligns closely with theoretical models that dictate morphological decomposition as a constant. These theories propose that words are often deconstructed into independent morphemes in receptive tasks and recognized that way, for example by deconstructing *jumping* into *jump* and *-ing*. At the other end of the spectrum stand theories that would place *jump* and *jumping* as separate lexical entries (Brysbaert & New, 2009, pp. 982–983).

Either way, there is strong evidence to suggest that inflected/derived forms and their base forms do affect each other in some way, suggesting that word families are a measure of a real representation in speakers' mental lexicon. In one such study, Nagy et al. (Nagy, Anderson, Schommer, Scott, & Stallman, 1989) explored the effect of both inflectional and derivational family frequency during a lexical decision task. They found that both types of morphological relationships lowered word recognition times, leading to the conclusion that inflections and derivational relationships are both represented in the mental lexicon, either through the grouping of related words under the same entry, or through linked entries. However, all the participants were native English speakers, so to what extent do L2 learners' lexicons reflect the same level of linking?

More recent studies have found that L2 learners' morphological knowledge and word-building ability are not nearly as developed. Ward and Chuenjundaeng -(Ward & Chuenjundaeng, 2009) conducted a study that tested the receptive ability of Thai engineering and doctoral students learning English. They were tested for their knowledge of a series of base words, together with various derived forms of the same words.

They found a surprising lack of familiarity with the derived words, even when participants knew the base forms from which they were derived. Similarly, but from a productive and not receptive standpoint, Schmitt and Zimmerman (2002) found that learners could produce only a limited number of derived forms when presented with a word family headword. These results challenge the common assumption that "once the base word or even a derived word is known, the recognition of other members of the family requires little or no extra effort" (Bauer & Nation, 1993, p. 253).

There is evidence (Mochizuki & Aizawa, 2000; Schmitt & Meara, 1997) to suggest a positive correlation between vocabulary size and morphological knowledge. If this is the case, then using higher-level word families in Bauer and Nation's framework for word list creation (as is the case in), may not be appropriate for learners with limited knowledge of vocabulary—the very learners that many of these lists target.

Similarly, a study by Jeon (2011) found that L2 learners' morphological knowledge leads to greater reading comprehension. Since many word lists are designed to increase reading comprehension in learners, it follows that they will likely be used by students without strong word-building abilities.

Clearly, then, when it comes to creating a word list, the unit of counting needs to fit the purpose and target audience of that list. Brezina and Gablasova (2015) contend that Bauer and Nation's (1993) higher word family levels ignore the lack of transparency that exists between many of the entries that would be placed under the same word family. Especially when creating a word list for beginners, Brezina and Gablasova point out that the morphological knowledge of language learners is often not developed enough. Because their New General Service List was created for beginners, and since it is intended to aid vocabulary acquisition for both receptive and productive purposes, Brezina and Gablasova chose the lemma as their unit of measure.

Seeking to quantify the effect of choosing to measure word families as opposed to word types, Sorell (2013) compared the text coverage of frequency lists made from the same four corpora. Each corpus corresponded to one of Sorell's text types (see *text types* above). Sorell's definition of "word families" was a slightly modified version of Bauer and Nation's (1993) sixth level of affix inclusion. He found, as would be expected, that the most frequent word families have a much larger text coverage than

the most frequent types. This is especially true when measuring type coverage—the most frequent word families accounted for roughly 4–6 times as many types in each corpus. However, when measuring overall token coverage, the top word families only covered about 3–10% more than the same number of most frequent types. Sorell also found that the most frequent 1,000 word families consisted of 6,557 word types in the general writing corpus. The number was similar in the other text types, though somewhat lower.

2.2.3 Objective design

Many word lists—including some of the most widely-known lists—take what could be termed a semi-objective approach. They begin by creating a list that bases word rankings on statistical measures such as frequency, range, and dispersion. Then, because certain words don't fit the researcher's intuitions, or because some rankings simply seem out of order, the list is tweaked here and there (Nation, 2016, p. 133).

For example, one common tweak is to group lexical sets together on a list, such as days of the week or numbers. This is true of West's GSL, resulting in a list that "brought a large element of subjectivity into the final product." (Brezina & Gablasova, 2015, p. 3) West himself laid out his argument as to why such an approach is preferable (1953, pp. ix–x).

Despite a few supposed pedagogical advantages, however, a semi-objective approach (which is therefore also a semi-subjective approach) has important implications for reproducibility. This alone makes it unfit for the present project, since one of the primary goals of this thesis is to present an easily reproducible process than can be use to create vocabulary lists in many different languages. Additionally, the simple fact is that by inserting subjective criteria into the list-creation process, it ceases to be based on the data directly. Rather than letting a particular corpus speak for itself, the whims and opinions of the researcher come into play. This can affect secondary tests that may be performed using the list, such as a lexical decision test.

Some lists that use strictly objective criteria include *Word Frequencies in Written and Spoken English* (Leech, Rayson, & Wilson, 2001), Brezine and Gablasova's *New General Service List* (2015), and Dang and Webb's *Essential Word List* (Nation,

2016, pp. 153–167). This thesis also uses exclusively objective criteria to create the *Conversational Hebrew Vocabulary List*: frequency, range, and dispersion. Let us now discuss each of these in turn.

2.2.3.1 Frequency Frequency can refer to either raw frequency (sometimes called absolute frequency) or normalized frequency. Raw frequency is simply the total number of times that a specific word is attested in the corpus. Normalized frequency is a measure of how many times the item appears *for every x tokens* in the corpus. This is usually calculated to be per-million-tokens, though the exact count can vary. Using normalized frequency is more meaningful since it is easier to compare with frequencies found in other corpora.

Frequency forms the core of frequency word lists, and it is also their most simple measure. A word list can be created using frequency alone. However, other measures, such as range, help take into account important factors that frequency ignores.

Gries (2010): > for example, observed frequencies (or their logs) are good proxies toward the familiarity of words—see Howes and Solomon (1951) for recognition times, Oldfield and Wingfield (1965) as well as Forster and Chambers (1973) for naming times, and Ellis (2002a, b) as well as Jurafsky (2003) and Gilquin and Gries (2009) for overviews.

2.2.3.2 Range Range is a measure of the number of sub-corpora—or sections of a corpus—in which the word can be found (Fries & Traver, 1960). Range is also sometimes referred to as *contextual diversity* (Brysbaert & New, 2009). To measure this, a corpus must first be divided into a series of sub-corpora. As of now, there is no real consensus on a specific way to do this, so different word lists may contain very different range measures based on the method chosen by the researcher. Like frequency, range can also be normalized to make the number more meaningful for inter-study comparison.

Nation has gone as far as to suggest that "range figures are more important than frequency figures, because a range figure shows how widely used a word is." (2016, p. 103) This conclusion is corroborated by studies such as that of Adelman, Brown,

and Quesada, which found that range better explained the findings of lexical decision tasks by 1%–3% (Adelman, Brown, & Quesada, 2006). Similar results were found by Ellis, who attributed better predictive power to range than to word frequency (2002a, 2002b).

The value of calculating range is that it provides a simple way to evaluate skewed frequency results. For example, a word may be rare overall in a language, but if it happens to be very common in only a few texts, it can still attain an inappropriately high place on the frequency list. This often occurs with specialized words that are only used by a very specific subset of the population but with high frequency. By calculating range, it becomes easy to identify these words.

The question then becomes, what to do once these words are found. How can range and frequency be used in tandem? One possibility, suggested by Nation and used by , is to decide on a minimum range, discard any words that fall below this bar, and order only the remaining words by frequency. This approach, however, relies on a subjective decision that becomes diffcult to replicate with other corpora. The fate of words with range close to the cutoff point is to be either completely thrown out or kept in their original position. Shifting the word's position on the list—its rank—is more sensical, but this can quickly become messy and subjective as well. Dispersion tries to solve this problem.

2.2.3.3 Dispersion In a (simplistic) nutshell, dispersion is a combination of both frequency and range. It serves as a single number—a distributional statistic—that incorporates the benefits of both of these measures, while also allowing a list to be ranked in a methodical, objective manner.

Unfortunately, there is still little agreement on how best to measure dispersion. Many ideas have been proposed, such as Juilland's D (Juilland, Brodin, & Davidovitch, 1970), Carroll's D_2 (1970), Rosengren's S (1971), Lyne's D_3 (1985), and Zhang's *Distributional Consistency* (DC) (Zhang, Huang, & Yu, 2004). A thorough overview of these and other dispersion measures was published by Gries, who then provided his own suggested method, *deviation of proportions*, or DP (2008, 2010; Lijffijt & Gries, 2012).

Unlike earlier proposals, however, Gries' *DP* stands out as a comparatively simple calculation that takes into account some of the biggest problems he identified in the others. Gries himself lists the benefits of *DP* as: flexibility to use differently sized subcorpora, simplicity, extendability to different scenarios, and appropriate sensitivity.

Brezina and Gablasova (105), p. 8: > ARF is a measure that takes into account both the absolute frequency of a lexical item and its distribution in the corpus (Savicky´and Hlava´c ˇ ova´2002; Hlava´c ˇ ova´2006). Thus if a word occurs with a relatively high absolute frequency only in a small number of texts, the ARF will be small (cf. Cerma´k and Kr ˇ en 2005; Kilgarriff 2009). All four wordlists were then sorted according to the ARF that ensured that only words that are frequent in a large variety of texts appeared in the top positions in the wordlists.

Sorell (2013), p. 89: Dispersion.

3 Methods: Creating the Conversational Hebrew Vocabulary List (CHVL)

As we have seen, the brunt of the work in high-quality vocabulary frequency list creation has focused on *English* frequency lists. Outside of the English-speaking world, and especially when dealing with less commonly taught languages, it's difficult to find well-researched word lists, if they exist at all. Why have not more educators—those who may benefit from these lists the most—decided to undertake such a task?

This need not be a project that one starts from scratch every time. Many tools already exist to make the process smoother. Still, with the rapid pace at which technology changes, these tools tend to quickly become obsolete. They are also usually restrictive to the specific preferences of their creators.

Rather than using these tools, I chose to create a series of simple scripts to create the Conversational Hebrew Vocabulary List.

The two most widely-used languages for the type of data analysis involved in a word list creation are Python and R. I chose to use Python for this project. Python was designed specifically to be a very readable programming language. That is, it is easy to read and understand the purpose and flow of the code. This was one of my primary reasons for choosing to use it, since it increases the ease with which this project can be reproduced by other researchers and educators to create their own word lists. R, on the other hand, requires a deeper familiarity with the syntax and conventions of the language in order to understand.

The second characteristic that makes Python ideal for an open-source project of this nature is its mild learning curve. Though considerable effort must be made to learn any programming language, Python is widely considered good for beginners because of its simplicity. With only a rudimentary knowledge of Python, even educators or enthusiasts without a coding background will be able to modify the scripts used here to suit their own needs. To this end, I will also carefully explain what, exactly, the code does.

Though all of the code is included in this thesis (*Appendix B*), it can also be found

in an online repository at https://github.com/juandpinto/opus-lemmas. The repository can easily be cloned, or individual files can be downloaded, for modification and use. The repository uses the version control system *Git*. This means that anyone can easily look through the history of each file to see specific changes that have been made over time.

Suggestions for improvements can also be submitted through the GitHub interface, allowing for a system of cooperation and incremental innovation among researchers. The exported Conversational Hebrew Vocabulary List, in its entirety, can also be found in the repository.

This thesis, then, beyond explaining the theory behind the creation of the CHVL, aims to make the process as reproducible as possible. This section contributes to that aim by carefully documenting each step of the process.

3.1 The corpus

Before coding or analyzing anything, it's important to find an appropriate corpus to use and to become familiar with its structure. A useful place to begin is OPUS[4], which is part of the Nordic Language Processing Laboratory (NLPL), and hosted by the CSC IT center in Finland. OPUS is a database of many open, parallel corpora. These include corpora of movie and television subtitles, TED talks, web-crawled data, newspapers, and of course, books. The corpora are all free and open to the public.

The CHVL was created using one of OPUS's corpora, the OpenSubtitles2018[5] corpus. The corpus can be downloaded in a variety of formats, and can be downloaded either as *parallel* corpora, or as a monolingual corpus. A parallel corpus consists of two languages interwoven together. For example, a line from the English subtitles of a movie will be paired with the same line from the French subtitles of the same movie. In theory, this means that each line of the corpus should have the same meaning in two different languages. The creation of parallel corpora has made possible many interesting and useful tools for linguistics, translators, and language learners. These

[4]http://opus.nlpl.eu
[5]http://opus.nlpl.eu/OpenSubtitles2018.php

include the open-source CASMACAT[6] project and the ReversoContext[7] tool.

For the purpose of creating a word list, a monolingual corpus is best. Note that parallel corpora will often be composed of less tokens than monolingual ones. This is because parallel corpora will only include movies for which the subtitles exist in both selected languages.

Though it's possible to download plain text files, the most useful format available for download is XML. Indeed, the most common file format used for large corpora is XML. The XML structure allows for nested key-value pairs, which are especially useful for parsed corpora that contain extensive metadata. XML is comparable to JSON, which we will use later to extract specific movie metadata directly from a database.

Another factor to consider is whether to download an untokenized, tokenized, or parsed corpus. An untokenized corpus contains simply the raw lines of text as found in the original subtitle files (divided into lines as they would appear while watching the movie, and labeled with the appropriate time for them to be shown):

```
<s id="49">
  <time id="T39S" value="00:03:22,280" />
שרלוק, אומר אתה מה?
  <time id="T39E" value="00:03:24,120" />
</s>
```

A tokenized corpus has further been split into individual words and punctuation, such that each word is tagged on its own:

```
<s id="49">
  <time id="T39S" value="00:03:22,280" />
  <w id="49.1">מה</w>
  <w id="49.2">אתה</w>
  <w id="49.3">אומר</w>
```

[6] http://www.casmacat.eu
[7] http://context.reverso.net/translation/

```
  <w id="49.4">,</w>
  <w id="49.5">שרלוק</w>
  <w id="49.6">?</w>
  <time id="T39E" value="00:03:24,120" />
</s>
```

A parsed corpus contains much more information for each token. The data included depends on the features of the language and on the parsing script used, but it can include things such as part of speech, syntactic role, lemma, and even specific features like gender, person, and number. Here is an example:

```
<s id="49">
  <time value="00:03:22,280" id="T39S" />
  <w xpos="ADV" head="49.3" feats="PronType=Int" upos="ADV"
  ↪  lemma="מה"
     id="49.1" deprel="obj">מה</w>
  <w xpos="PRON" head="49.3" feats="Gender=Masc|Number=Sing|Person=2|
     PronType=Prs" upos="PRON" lemma="הוא" id="49.2"
↪  deprel="nsubj">אתה</w>
  <w xpos="VERB" head="0"
  ↪  feats="Gender=Masc|HebBinyan=PAAL|Number=Sing|
     Person=1,2,3|VerbForm=Part|Voice=Act" upos="VERB"
↪  misc="SpaceAfter=No"
     lemma="אמר" id="49.3" deprel="root">אומר</w>
  <w xpos="PUNCT" head="49.3" upos="PUNCT" lemma="," id="49.4"
     deprel="punct">,</w>
  <w xpos="NOUN" head="49.3" feats="Gender=Masc|Number=Sing"
  ↪  upos="NOUN"
     misc="SpaceAfter=No" lemma="שרלוק" id="49.5"
↪  deprel="obj">שרלוק</w>
  <w xpos="PUNCT" head="49.3" upos="PUNCT" misc="SpaceAfter=No"
  ↪  lemma="?"
     id="49.6" deprel="punct">?</w>
```

```
  <time value="00:03:24,120" id="T39E" />
</s>
```

All of the data used to create the CHVL came from a monolingual parsed corpus of Hebrew. The parsing was all done automatically using .

3.2 Cleansing the corpus

Unlike many corpora, the OpenSubtitles2018 corpus as presented in its downloadable form has already undergone significant preprocessing by the OPUS team.(Lison & Tiedemann, 2016) This is good news, since data cleansing is often the most laborious part of the process. However, there is one issue that must be addressed before the corpus can be used to create a word list: deduplication.

The files inside the downloaded folder are organized as follows:

```
Zipped folder in GZ format
    Folder for year X
        Folder for movie A
            Zipped XML in GZ format
            Zipped XML in GZ format
            Zipped XML in GZ format
        Folder for movie B
            Zipped XML in GZ format
            Zipped XML in GZ format
    Folder for year Y
        Folder for movie C
            Zipped XML in GZ format
        Folder for movie D
            Zipped XML in GZ format
            Zipped XML in GZ format
            Zipped XML in GZ format
        Folder for movie E
```

```
        Zipped XML in GZ format
        Zipped XML in GZ format
Folder for year Z
      Folder for movie F
          Zipped XML in GZ format
          Zipped XML in GZ format
```

This organization is straight-forward, except for the fact that there are multiple XML files for each movie. The subtitle files that OPUS has collected, parsed, organized, and made available for mass download were all obtained from the Open Subtitles[8] project (hence the name of the corpus). Because this is a database where users can upload the subtitle files they extract from their own movie collection, there are often multiple uploads for the same movie. For our purposes, this results in movies that can have anywhere from a single subtitle file to dozens of them. Unfortunately, though the tokens in the files themselves are usually the same (with only minor variations in the XML metadata), this is not always true. Some few variations seem to be different and independent translations.

Part of cleansing the corpus, then, entails getting rid of these duplicates. As a means of simplifying the entire process, I chose simply to use the first file in each movie folder. I've included the short Python script for this in its entirety in *Appendix B.3*. However, I will here explain what it does in detail so that it can be easily modified to fit different circumstances.

The script first makes a copy of the entire folder structure in the original downloaded (and unzipped!) corpus into a new directory. It then finds the first XML file in each movie folder and copies it into the appropriate place in the new folder structure. This means that it doesn't delete or otherwise change the files in the original corpus in any way.

The first block of code imports necessary modules that are used later in the script (`shutil` and `os`). Lines 7 and 8 define where the original corpus is (`source`), and where the new one will be placed (`destination`).

[8]

```
import shutil
import os

source = '../OpenSubtitles2018_parsed'
destination = './OpenSubtitles2018_parsed_single'
```

Next, a single line of code copies all directories and subdirectories into their new location.

```
shutil.copytree(source, destination,
↪  ignore=shutil.ignore_patterns('*.*'))
```

Lastly, we create a variable that holds all the XML files located in each movie folder, trim the list to just one, and copy that one into its new location. This process is carried out for one movie folder at a time. The originals are left untouched.

```
for dirName, subdirList, fileList in os.walk(source):
    for fname in fileList:
        if fname == '.DS_Store':
            fileList.remove(fname)
    if len(fileList) > 0:
        del fileList[1:]
        src = dirName + '/' + fileList[0]
        dst = destination + dirName[27:] + '/'
        shutil.copy2(src, dst)
```

With a newly organized version of the corpus, it's now possible to begin the process of reading and processing data. At this stage, I took some time to gather metadata for all the movies in the corpus in order to identify movies that were originally filmed with Hebrew as their primary language (as opposed to translated subtitles). Because I ultimately decided against this approach for the creation of the CHVL, I will skip that step here. However, a description of the entire process will be discussed later under *Using original-language movies exclusively.*

3.3 Reading data

Before calculating any measures such as frequency, individual lemmas must be extracted from the XML files in the downloaded corpus. There are two ways to go about this. Because XML consists of nested tags and key-value pairs, a dedicated XML parsing tool can be used to extract specific information. In this case we would be creating a list of all *values* in the `'lemma'` *key* within each `<w>` *tag*. The value that corresponds to the `'lemma'` tag below for the word אומר is אמר.

```
<w xpos="VERB" head="0"
↪  feats="Gender=Masc|HebBinyan=PAAL|Number=Sing|
   Person=1,2,3|VerbForm=Part|Voice=Act" upos="VERB"
↪  misc="SpaceAfter=No"
   lemma="אמר" id="49.3" deprel="root">אומר</w>
```

A different approach is to use *regular expressions* to search for a specific string of characters and extract every instance of that string. This is a more brute-force approach, since it ignores the structure of the XML file and treats it all simply as raw text. To find a lemma, a very simple regular expression is sufficient: `lemma="[א-ת]+"`. This will search for any instance of the characters `lemma="`, followed by a combination of any number of Hebrew letters (at least one), followed by the character `"`.

Despite the existence of various Python modules for parsing XML files, I found a simple search using regular expressions to be more efficient for various reasons. First, not all elements in the parsed corpus contain *lemma* attributes. Second, punctuation and non-Hebrew words are often lemmaticized. This means that even after extracting all the *lemma* values in a file, I would still need to use regular expressions to search through the results and delete any that contain non-Hebrew characters. I chose instead to skip the XML parsing step altogether.

I will now explain the code in the script used to create the CHVL. As with the other code, the entire script in its entirety can be found in *Appendix B.1.*

After importing necessary packages and initializing variables, two functions near the beginning of the script serve to open a file and extract a list of lemmas from it.

```
# Open XML file and read it.
def open_and_read(file_loc):
    with gzip.open(file_loc, 'rt', encoding='utf-8') as f:
        read_data = f.read()
    return read_data
```

```
# Search for lemmas and add counts to "lemma_by_file_dict{}".
def find_and_count(doc):
    file = str(f)[40:-3]
    match_pattern = re.findall(r'lemma="[א-ת]+"', doc)
    for word in match_pattern:
        if word[7:-1] in lemma_by_file_dict:
            count = lemma_by_file_dict[word[7:-1]].get(file, 0)
            lemma_by_file_dict[word[7:-1]][file] = count + 1
        else:
            lemma_by_file_dict[word[7:-1]] = {}
            lemma_by_file_dict[word[7:-1]][file] = 1
```

We then run both of these functions for each XML file in the corpus directory (defined earlier in `corpus_path`).

```
for dirName, subdirList, fileList in os.walk(corpus_path):
    if len(fileList) > 0:
        f = dirName + '/' + fileList[0]
        find_and_count(open_and_read(f))
```

The `find_and_count()` function finds each instance of the string described above using a regular expression, then adds the Hebrew part of the string—the lemma itself—to a dictionary. The dictionary is named `lemma_by_file_dict`, and its structure looks like this:

```
'lemma': {'path of file': 'frequency of lemma in file'}
```

A dictionary is at its core a list of key:value pairs. Much like an actual dictionary consists of words and their definitions, this dictionary's keys are made up of all the individual lemmas found by our search. For each lemma, the value is another dictionary—making it a nested dictionary, or a dictionary within a dictionary. The keys for each inner dictionary are the paths of all the XML files (movies) that the lemma appears in, and the value of each is an integer that represents how many times that lemma appears in that file (frequency).

After the script reads each file, it returns a complete dictionary. Here is a sample:

```
'ב': {
        '/he/0/5753574/6853341.xml': 168,
        '/he/0/3607000/5764778.xml': 94},
'פרק': {
        '/he/0/5753574/6853341.xml': 3},
'קודם': {
        '/he/0/5753574/6853341.xml': 6,
        '/he/0/3607000/5764778.xml': 2,
        '/he/0/1278351/3777598.xml': 1}
```

Throughout the rest of the script, this nested dictionary serves as the basis for all of the calculations needed.

3.4 Calculations

For each lemma, the CHVL includes three measures: frequency, range, and U_{DP} (dispersion). It uses dispersion as its sorting value. Let's look at how each of these is calculated. Range will be addressed in the export section, since the script calculates it on the spot as the list is created.

3.4.1 Frequency

Since we've already calculated the frequency of each lemma for each individual file, calculating total frequency per lemma is straight forward. The script simply creates a new dictionary, `lemma_totals_dict`, and adds to it every lemma in the corpus as its keys, with the corresponding value being a sum of the frequencies in all files for that lemma. In other words, {'lemma1':'frequency1', 'lemma2':'frequency2', . . . }

```
116 for lemma in lemma_by_file_dict:
117     lemma_totals_dict[lemma] =
 ↪  sum(lemma_by_file_dict[lemma].values())
```

This returns Using the short example given above, this would result in the following dictionary:

```
262:'ב',
3:'פרק',
9:'קודם'
```

3.4.2 U_{DP} (dispersion)

Dispersion is more complicated. In theory, it should provide a single quantifiable measure that incorporates both frequency and range, and which can then be used to sort the word list. There is no agreed-upon, single way to calculate dispersion, and different researchers will use the words in slightly different contexts. The model of dispersion I have chosen to follow for this project is Gries' dispersion coefficient, or U_{DP}, () calculated from Gries' DP. ()

In order to calculate Gries' DP for $lemma_x$, we must first make two calculations for each file in the corpus ($file_i$): the lemma's *expected frequency* if it were perfectly distributed, and its *observed frequency*—or its actual frequency.

$$\textbf{expected frequency} = \frac{tokens\ in\ file_i}{tokens\ in\ corpus}$$

$$\textbf{observed frequency} \;=\; \frac{frequency\ of\ lemma_x\ in\ file_i}{frequency\ of\ lemma_x\ in\ corpus}$$

We must then subtract the lemma's observed frequency from its expected frequency, which will return a value between -1 and 1. We can normalize this result by finding the absolute value. Now the closer the result is to 0, the closer that lemma's frequency is in that particular file to what we would expect if it were perfectly distributed throughout the corpus. A higher number (closer to 1), would indicate a heavier load in that file that we would expect.

By performing this calculation for every file in the corpus, adding them all together, and dividing the result by two (since we're using the absolute value and are therefore adding values originally in both directions), we now have Gries' DP. Where `n` is the number of files:

$$\textbf{DP} \;=\; 0.5 \sum_{i=1}^{n} |\ \text{expected frequency} \;-\; \text{observed frequency}\ |$$

A DP of 0 represents a perfectly even dispersion, and a DP close to 1 means a more uneven distribution, where fewer files contain a larger load of the lemma's overall frequency. A DP of 1 is not actually possible.

Gries' usage coefficient, or U_{DP}, is an attempt to make this number more useful. DP is first subtracted from 1 and the result is multiplied by the lemma's total frequency. The full equation for U_{DP} is as follows:

$$\left(1 - 0.5 \sum_{i=1}^{n} \left| \frac{file_i\ tokens}{total\ tokens} \;-\; \frac{frequency_x\ in\ file_i}{total\ frequency_x} \right| \right) \times total\ frequency_x$$

In order to calculate this, the script must first find the number of tokens in each file. Like before, this is done by creating a dictionary, `token_count_dict`, which contains the key:value pairs of file:tokens. Since we already have a dictionary with the number of times that each lemma appears in each file, `lemma_by_file_dict`, we don't need to open and read the files again. Instead, we can add the values in this

dictionary and rearrange them into what we want.

```
120 for lemma in lemma_by_file_dict:
121     for file in lemma_by_file_dict[lemma]:
122         token_count_dict[file] = token_count_dict.get(
123             file, 0) + lemma_by_file_dict[lemma][file]
```

We also need to know the total number of tokens in the entire corpus. This is a simple matter of adding all the values in the `token_count_dict` dictionary. The final count is saved into an integer variable, `total_tokens_int`.

```
126 for file in token_count_dict:
127     total_tokens_int = total_tokens_int + token_count_dict.get(file,
    ↪  0)
```

Finally, the script uses all these measures to calculate DP and then U_{DP} for each lemma, and places them into their respective dictionaries, `lemma_DPs_dict` and `lemma_UDPs_dict`.

```
129 # Calculate DPs
130 for lemma in lemma_by_file_dict.keys():
131     for file in lemma_by_file_dict[lemma].keys():
132         lemma_DPs_dict[lemma] = lemma_DPs_dict[lemma] + abs(
133             (token_count_dict[file] /
134              total_tokens_int) -
135             (lemma_by_file_dict[lemma][file] /
136              lemma_totals_dict[lemma]))
137 lemma_DPs_dict = {lemma: DP/2 for (lemma, DP) in
    ↪  lemma_DPs_dict.items()}
138
139 # Calculate UDPs
140 lemma_UDPs_dict = {lemma: 1-DP for (lemma, DP) in
    ↪  lemma_DPs_dict.items()}
```

With these values all calculated for each lemma, the only thing left is to sort and create the final list.

3.5 Sort and export

In order to ensure that the words on the list do not have an abnormally high frequency in some subcorpora (movies) and are nearly absent in others, some have suggested setting a minimum range or dispersion. All words that fall below this threshold are discarded, and the remaining words can then be sorted by frequency.

Though this is a more systematic approach than that used to create many early frequency lists, it still depends on a subjective decision and the whim of the researcher.

Rather than setting an arbitrary bar, the CHVL is sorted entirely by Gries' usage coefficient of dispersion (U_{DP}). This *modus operandi* ensures that the order of words itself—not just which words make it onto the list and which don't—is decided by a combination of both relevant measures: frequency and dispersion. This approach also has the added benefit of being entirely objective.

Since we've already calculated the U_{DP} for each lemma, sorting the list is simple.

```
148 UDP_sorted_list = [(k, lemma_UDPs_dict[k]) for k in sorted(
149     lemma_UDPs_dict, key=lemma_UDPs_dict.__getitem__,
150     reverse=True)]
```

A final table is then created (using a list of tuples, `table_list`), with each line consisting of a lemma, its overall frequency, its range, and its U_{DP}. This table is already sorted by U_{DP} as it's being created.

Because the script has not calculated range by this point, it must do so on the spot as it's entering each lemma into the table. It does this with a simple dictionary comprehension that quickly counts the number of files included in the `lemma_by_file_dict`. Here is the resulting code:

```
153 for k, v in UDP_sorted_list[:list_size_int]:
154     table_list.append((k, lemma_totals_dict[k], sum(
155         1 for count in lemma_by_file_dict[k].values() if count > 0),
156         v))
```

Lastly, now that everything is organized into a table, the script opens (or creates, if it doesn't yet exist) a CSV file, writes a header line into it (`LEMMA, FREQUENCY, RANGE, UDP`), and exports the entire table into the file. It then closes it to clear the computer's memory cache.

```
199 result = open('./export/WordList.csv', 'w')
200 result.write('LEMMA, FREQUENCY, RANGE, UDP\n')
201 for i in range(list_size_int):
202     result.write(str(table_list[i][0]) + ', ' +
203                  str(table_list[i][1]) + ', ' +
204                  str(table_list[i][2]) + ', ' +
205                  str(table_list[i][3]) + '\n')
206 result.close()
```

The list is now complete. The next section will explore the list itself more in-depth.

4 The CHVL: A vocabulary list of conversational Modern Hebrew

The Conversational Hebrew Vocabulary List in its entirety can be found as an electronic supplement to this thesis (in CSV format) or at the following GitHub repository: *https://github.com/juandpinto/opus-lemmas*. It contains the most common 5,000 lemmas of conversation Modern Hebrew, as found in the OpenSubtitles2018 corpus. A sample of the first 1,000 lemmas is included in *Appendix A*.

For discussion purposes, a small sample of the first 30 items is here presented.

Table 1: Sample of the first 30 items on the CHVL.

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
1	הוא	121,008.92	43455	22,227,310.52
2	ה	50,841.12	43458	9,153,952.58
3	את	35,337.28	43426	6,357,357.64
4	ל	29,102.77	43448	5,311,835.36
5	לא	27,213.76	43433	4,822,345.74
6	זה	26,418.69	43441	4,614,840.01
7	ב	24,839.48	43450	4,472,208.92
8	של	20,088.89	43445	3,529,189.96
9	ש	20,028.64	43439	3,527,087.63
10	היה	13,312.52	43420	2,298,194.02
11	מה	12,192.80	43403	2,107,876.08
12	ו	9,840.85	43429	1,687,960.58
13	על	9,119.70	43430	1,597,865.21
14	כול	6,842.01	43414	1,174,558.76
15	ידע	6,205.85	43323	1,032,405.06
16	כן	6,232.26	43226	971,073.85
17	מ	5,479.15	43411	943,781.99
18	יש	5,519.12	43376	937,885.08
19	עשה	4,941.68	43311	810,088.75
20	אבל	4,757.33	42963	785,248.37

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
21	טוב	4,891.35	43291	766,201.25
22	רצה	4,671.67	43202	765,197.00
23	אם	4,444.59	43321	745,301.07
24	עם	4,333.17	43331	727,755.37
25	אמר	4,128.07	43196	681,096.31
26	אז	4,052.24	43202	653,014.96
27	סדר	4,305.52	42733	619,555.39
28	צריך	3,501.64	43101	554,553.56
29	רק	2,996.30	43306	492,899.21
30	חשב	3,021.85	43062	486,623.93

Besides each lemma and its respective rank on the list, the CHVL includes three pieces of information: frequency, range, and U_{DP}. Frequency in this case is not raw frequency—the total number of times the lemma appears in the corpus—but rather how many times the lemma appears for every million tokens in the corpus. Using this normalized frequency measure makes the number more meaningful since it aims to reflect the per-million count of all spoken Hebrew, not just the OpenSubtitles2018 corpus. It also makes it easier to compare frequencies with those found in other corpora. The range is the number of sub-corpora—or in this case, movies—the lemma appears in.

The most important piece of information the list provides, however, is the U_{DP}, which refers to Griers' usage coefficient for dispersion. This is discussed more in-depth in the *methods* section above. U_{DP} is also used as the sorting measure for the CHVL.

The percentage of the corpus that is covered by the first n items on the list is referred to as coverage. This is a simple matter of finding the total number of tokens in the corpus, and dividing from it the sum of all the *raw* frequencies from the first n items.

For example, the sum of the frequencies of the first 20 lemmas in *Table 1* (84,656,819) divided by the total size of the corpus (193,755,220) is 0.436926649. In theory, this means that by knowing just the first 20 lemmas on the CHVL one would be able to understand 43.7% of the words in the entire OpenSubtitles2018 corpus! That is a

clear example of the power of Zipf's Law (see *Introduction* for more on Zipf's Law).

Table 2 presents a listing of some important coverages provided by different amounts of lemmas on the CHVL.

Table 2: Breakdown of coverage percentages.

n Lemmas	Frequency Sum	÷ Corpus Size	= Coverage
374	135,767,644	193,755,220	70%
939	155,016,588	193,755,220	80%
4,246	174,380,519	193,755,220	90%
13,758	184,067,666	193,755,220	95%

The entire CHVL consists of 5,000 lemmas. This number was chosen in order for it to include the required items for 90% coverage, while also making it an even factor of 1,000. In its entirety, the CHVL covers 90.8% of the corpus from which it is created.

4.1 Challenges and future direction

Throughout the course of this project, I have encountered several issues that are worth discussing. Some of these are questions that require further study in order to address adequately. Others are technical issues related to the complex task of pre-processing and parsing the corpus—something not directly dealt with in this thesis. Others yet are simple suggestions that I simply did not have time to implement given this project's time constraints. And finally, there are limitations that are the inevitable result of the tools at hand.

I have divided all of these issues into two categories: methodological challenges of a bigger nature, and functional challenges of a more limited scope.

4.1.1 Methodological challenges

One of the more obvious issues of this project is the use of a corpus of movie subtitles as substitute for a corpus of true conversational language. This issue in a way forms

the backbone of the CHVL, and it is at the heart of what this project is all about. Though I discuss several points related to this in the *Background* section of this thesis, I will here discuss some of its implications for future work.

4.1.1.1 Ideal vs. practical corpora

The use of a subtitle corpus has both positive and negative aspects. As mentioned earlier, the early research that has been done on the topic indicates that movie subtitles share many features with spontaneous, spoken language. This includes a high level of correlation between the two , as well as a strong ability to predict the outcomes of a lexical decision task .

One especially positive aspect of subtitle corpora is their accessibility. Thanks to the efforts of organizations such as *http://opensubtitles.com* and OPUS[9], very large corpora are available to the public for free. And they already come pre-processed, as an additional incentive for the time-constrained researcher.

This free and open nature makes subtitle corpora excellent tools for research in languages that don't yet have large, high-quality corpora of spoken language. Though advances in technology are rapidly making this type of data-collection more accessible, the costs remain too high for many less-commonly taught languages as of now. This is largely due to the arduous process of transcribing audio recordings.(Izre'el, 2004)

An ideal corpus for this sort of task would consist of many millions of tokens of recorded, transcribed, and parsed, spontaneous spoken language. Several attempts have been made to create a corpus of this nature in Hebrew.

The most prominent of these is the Corpus of Spoken Israeli Hebrew (CoSIH)[10], created at Tel Aviv University between 2000 and 2002.(Izre'el, Hary, & Rahav, 2001) Designed and initiated by a team of distinguished scholars, it unfortunately ran out of funding long before its goals were met. The CoSIH website (*http://cosih.com/*) makes available to the public a total of 13.5 hours of recorded Hebrew, with just over five hours of it having been transcribed.

Though a few publications have used data from CoSIH, these have been primarily

[9] http://opus.nlpl.eu
[10] http://cosih.com/

methodological studies for the design of the project itself.(Amir, Silber-Varod, & Izre'el, 2004; Izre'el et al., 2005; Mettouchi, Lacheret-Dujour, Silber-Varod, & Izre'el, 2007) At least one dissertation, by Nurit Dekel, uses data exclusively from CoSIH. Her entire corpus consists of 44,000 tokens. (2010, p. 7)

Other corpora of spoken Hebrew include the Haifa Corpus of Spoken Hebrew (Yael, 2014) and the Hebrew CHILDES corpus (Albert, MacWhinney, Nir, & Wintner, 2013; Gretz, Itai, MacWhinney, Nir, & Wintner, 2015). The first consists of 17.5 hours of audio recordings, along with a limited selection of transcribed text. The latter is a collection of recordings of interactions between adults and children, comprising a total of 417,938 transcribed tokens. The CHILDES corpus is unique in that the transcriptions are provided using a Latin-based phonemic transliteration. This was done in order to avoid many of the textual ambiguities of using the Hebrew script, which are addressed below under *functional challenges*.

Though ideal in some ways, these corpora remain far too small to be effectively used for the creation of frequency lists. Even combined into a single corpus (which would introduce a series of new issues to solve), the total size would not be bigger than two million tokens. As discussed earlier in this thesis, Sorell provides evidence to suggest that a corpus of 20–50 million tokens is the minimum for a stable word list.(2013)

Are movie and television subtitles an suitable substitute for spontaneous, spoken language? Early studies suggest it is at least adequate, but much more research is needed to answer this question definitively. For now, it remains as one practical option.

4.1.1.2 Using original-language movies exclusively One of the potential downsides of using the OpenSubtitles2018 corpus not yet discussed is that it includes all subtitles of a specific language, even *translated* subtitles from movies filmed in other languages. The question is, does a translated script represent true conversational language as faithfully as an original script?

This is a question that requires more research in order to answer satisfactorily. Though translated subtitles don't need to try to approximate the utterance length and visual cues that a dubbed script does, its quality still largely depends on the

skills of a translator. Most importantly, a translation may not accurately reflect the register of the original, no longer serving as a representation of conversational language. Again, these are important points to consider.

One solution is to simply use movies that were originally filmed in the target language of the corpus. In theory, each XML file in a monolingual OpenSubtitles2018 file should contain a tag that identifies the original language of the movie. In practice, I found that the overwhelming majority of the files contained an empty `<lang>` tag instead. Luckily, there is a way to obtain the desired metadata for each movie in the corpus.

This can be done with a script that uses an application programming interface (API) to fetch specific information from an online movie database. The name of each movie folder in the corpus, which is simply a series of numbers, corresponds to that movies IMDb ID, which is a unique ID registered with the Internet Movie Database[11]. This makes the process relatively easy, as we simply need to query the database using this ID to receive all of the movie's metadata.

Though IMDb does provide their own API, I decided instead to use an API created for the Open Movie Database (OMDb)[12]. This API can be used free-of-charge, but it has a 1,000 movie limit per day. Since the OpenSubtitles2018 Hebrew corpus contains nearly 50,000 movies, I decided instead to pay for a daily limit of 100,000 movies. This only requires a $1.00 donation for each month that one is registered to use the OMDb API.

Once an API key is obtained, a script can be written to obtain the information desired for every movie all at once. In this case, we want to know the original language(s) for each movie.

This script in its entirety is found in Appendix B.2. It uses an imported Python wrapper for the API, written by Derrick Gilland[13], which can be found at https://github.com/dgilland/omdb.py. This package can be installed through PIP by entering `pip install omdb` into the command line.

[11] http://www.imdb.com/
[12] http://www.omdbapi.com/
[13] https://github.com/dgilland

For practical purposes, the script requires one to enter a specific year (or, more accurately, corpus folder name). If desired, an asterisk can act as wildcard: `python OMDb-fetch.py 1988` will fetch data for movies from 1988, while `python OMDb-fetch.py 198*` will do it for all movies in the 1980s. In order to fetch data for all movies in the database at once, use `python OMDb-fetch.py *`. I don't recommend this, however, since it may overload the server and cause the script to time out.

The script begins by creating a list of all movie directory paths for the desired year.

```
15 for name in glob.glob(
16         './OpenSubtitles2018_parsed_single/parsed/he/' + year +
           ↪  '/*/'):
17     IDs.append(name)
```

Each item in the list is then trimmed to include only the name of the movie folder, which is *almost* equivalent to the IMDb ID.

```
20 IDs = [os.path.basename(os.path.dirname(str(i))) for i in IDs]
```

In order to make the IDs match those in the database, additional zeros must be added to the beginning until they are seven digits long.

```
23 for i in IDs:
24     while len(i) < 7:
25         IDs[IDs.index(i)] = '0' + i
26         i = '0' + i
```

The list is then sorted numerically in order to more easily interpret the results: `IDs.sort()`.

The API key is set in line 32, but be sure to replace `906517b3` with your own key, which can be obtained at http://www.omdbapi.com/.

```
32 omdb.set_default('apikey', '906517b3')
```

The script then prints a table header, fetches the title, year, and language(s) for each movie, and prints the results directly into the computer terminal.

```
35 print('# ' + year + '\n' +
36       'IMDb ID\tTitle\tYear\tLanguage(s)')
```

```
39       for i in IDs:
40           doc = omdb.imdbid('tt' + i)
41           print('tt' + i + '\t' +
42                 doc['title'] + '\t' +
43                 doc['year'] + '\t' +
44                 doc['language'])
```

4.1.2 Functional challenges

A quick scan of the CHVL reveals some notable items. Some of these are mere quirks of the automatic parser, while others are the result of ambiguities.

For example, the very first lemma on the list is a bit unexpected. "הוא" is certainly not the most common lemma in Modern Hebrew. A quick look at some of the files in the corpus, however, reveals that all pronouns are grouped under this lemma. That is, אתה (you), היא (she), and אנחנו (we), just to name a few, are parsed as belonging to the lemma "הוא." Considering how common pronouns are in the majority of spoken dialogue (in many languages), its place at the top of the list ceases to be a surprise.

Another thing to note is that verbs are all listed in their traditional third-masculine-singular past conjugation. The first verb on the list is "היה"—a lemma referring to all forms of the verb להיות, including the infinitive. The same is true of "ידע" (item 19) and "דיבר" (item 60).

Many of the most common lemmas on the CHVL are prepositions. Note that even inseparable prepositions, such as -ה and -ב are considered independent lemmas by the parser, and are listed respectively as the lemmas "ה" and "ב".

Other issues, however, are more difficult to explain.

4.1.2.1 Textual ambiguity of Hebrew orthography

The flexible spelling conventions of Hebrew are at the root of many of the problems with the CHVL. For example, דִּבֶּר *he spoke* can be written as either דיבר ("full spelling") or דבר ("defective spelling"). There is also a noun, דָּבָר *thing*, that looks identical to the verb's defective spelling (דבר). Though the difference is usually clear from context, the automatic parser has some difficulty with this orthographic ambiguity.

The lemma "דבר" (item 27) includes instances of both the verb and the noun, which are completely unrelated. A simple search through the corpus reveals multiple examples of the noun דבר tagged with `lemma="דבר"`:

```
<w xpos="NOUN" head="579.3" feats="Gender=Masc|Number=Sing"
 ↪  upos="NOUN" lemma="דבר" id="579.2" deprel="nsubj">דבר</w>

<w xpos="NOUN" head="200.11" feats="Gender=Masc|Number=Plur"
 ↪  upos="NOUN" lemma="דבר" id="200.12" deprel="obj">דברים</w>
```

We also find plenty of examples of the verb with the same lemma tag:

```
<w xpos="VERB" head="0"
 ↪  feats="Gender=Fem|HebSource=ConvUncertainHead|Number=Sing|Person=3|Tense=Past"
 ↪  upos="VERB" lemma="דבר" id="2346.4" deprel="root">דברה</w>

<w xpos="VERB" head="0"
 ↪  feats="Gender=Fem,Masc|Number=Plur|Person=1|Tense=Past"
 ↪  upos="VERB" lemma="דבר" id="1270.2" deprel="root">דברנו</w>

<w xpos="VERB" head="0"
 ↪  feats="Gender=Fem,Masc|Number=Plur|Person=3|Tense=Past"
 ↪  upos="VERB" lemma="דבר" id="368.4" deprel="root">דברו</w>
```

A different lemma, "דיבר" (item 61), is the expected lemma for the verb since it follows the standard third masculine plural conjugation. Interestingly, however, the parser applies this lemma only to attestations of the word with an inserted *yod*, or with a *mem* or *lamed* prefix (present tense or infinitive). All other instances are parsed as the lemma "דבר." Though unexpected and simply wrong, at least this issue is consistent.

```
<w xpos="VERB" head="840.4"
↪  feats="Gender=Fem,Masc|HebBinyan=HITPAEL|Number=Plur|Person=1|Tense=Past"
↪  upos="VERB" lemma="דיבר" id="840.16" deprel="conj">דיברנו</w>

<w xpos="VERB" head="1451.12"
↪  feats="Gender=Masc|HebBinyan=PIEL|Number=Sing|Person=1,2,3|VerbForm=Part|Voice=A
↪  upos="VERB" lemma="דיבר" id="1451.20" deprel="obl">מדבר</w>
```

To complicate matters more, we also find the unexpected lemmas "דיברה" (item 1184), "שדיבר" (item 2588), and "שדיברה" (item 4106).

Which, based on context (), should clearly be parsed as two separate lemmas, "ש" and "דיבר."

These are just a few among many examples of the difficulties encountered by the automatic parser. Though the parsing was carried out by the OPUS team as part of the corpus's pre-processing stage, it is valuable to at least have an idea of how it works its magic. I will here explain the basics of the process and some of the implications entailed.

4.1.2.2 Automatic parsing Automatic parsing refers to the process of having a computer program create a syntactic tree for a corpus of natural language. Natural language, as opposed to artificial or constructed language, is notoriously complex in its structure. Natural language processing (NLP) is an entire field of research, currently at the forefront of computer science. Parsing can serve many purposes, from

theoretical linguistic research to machine translation or even the creation of artificial intelligences such as Siri or Alexa. For our purposes, a parsed text is important in order to use lemmas as the word family level for the CHVL. This decision is discussed under *Identifying Words* in this thesis.

Two distinct types of syntactic parsers exist, contituency parsers and dependency parsers. These are based on the two respective linguistic theories of syntax, constituent grammar (sometimes referred to as phrase structure grammar) and dependency grammar.

Constituent grammar is the classic syntax tree structure taught in introductory-level linguistics classes. It is essentially a theory of the logic structure of language as a whole. Dependency grammar is a competing theory that treats words as more directly interconnected to each other. A thorough description of these ideas is outside the scope of this thesis, and is not pertinent to the project. What is important to know is that dependency grammar, and thus dependency parsers, have played an important role in the advancement of NLP and computational linguistics as a whole. The term "automatic parser", therefore, most often refers to an automatic *dependency* parser.

Some parsers proceed in a two-step process of morphological tagging (part of speech) and then dependency parsing (syntactic role and conjugations). In all cases, tokenization must first take place, which refers to splitting the text into individual lemmas.

Most automatic parsers are "trained" using a small corpus that has been manually parsed by a human previously, or at least one that was automatically parsed and then checked and corrected by the researcher. These "gold-standard" pre-parsed corpora are called treebanks, and repositories of them they have been created for many languages. Building on existing databases of knowledge, these many of these parsers use statistical models to determine the most likely syntactic structure and conjugation for each word in each sentence.

Some parsers, however, are instead simply given entirely unparsed corpora and no knowledge of the language's syntactic structure. Working with nothing but the text itself, the program seeks out patterns and begins to create links and relationships that it deems significant.

Unfortunately, though automatic parsers have achieved surprising levels of accuracy in recent years, even the best continue to produce erroneous parsings. Some researchers have claimed as 95% or higher accuracy, including for some Hebrew parsers. When dealing with such a large corpus, such as the Hebrew OpenSubtitles2018 corpus consisting of nearly 200 million tokens, a best-case scenario for a 5% error threshold results in nearly 10 million incorrectly parsed words.

Undoubtedly, this can have a negative impact on the accuracy of lemma frequency counts. Many of the issues found in the CHVL are not due to orthographic ambiguity, but simply to inaccurate parsing. Some, as shown in the previous section, are even caused by erroneous automatic tokenization (consider the lemma "שדיבר").

The good news is that automatic parsers are continually improving in accuracy. This is a problem that exists across the board, regardless of the corpus being used—unless it is manually parsed and lemmaticized, which is nearly impossible for such large corpora. The tools and techniques outlined in this thesis do not directly deal with the process of parsing.

5 Implications for other less commonly taught languages

5.1 Easy reproducibility and growth

Appendix A: Conversational Hebrew Vocabulary List (CHVL)

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
1	הוא	121,008.92	43455	22,227,310.52
2	ה	50,841.12	43458	9,153,952.58
3	את	35,337.28	43426	6,357,357.64
4	ל	29,102.77	43448	5,311,835.36
5	לא	27,213.76	43433	4,822,345.74
6	זה	26,418.69	43441	4,614,840.01
7	ב	24,839.48	43450	4,472,208.92
8	של	20,088.89	43445	3,529,189.96
9	ש	20,028.64	43439	3,527,087.63
10	היה	13,312.52	43420	2,298,194.02
11	מה	12,192.80	43403	2,107,876.08
12	ו	9,840.85	43429	1,687,960.58
13	על	9,119.70	43430	1,597,865.21
14	כול	6,842.01	43414	1,174,558.76
15	ידע	6,205.85	43323	1,032,405.06
16	כן	6,232.26	43226	971,073.85
17	מ	5,479.15	43411	943,781.99
18	יש	5,519.12	43376	937,885.08
19	עשה	4,941.68	43311	810,088.75
20	אבל	4,757.33	42963	785,248.37
21	טוב	4,891.35	43291	766,201.25
22	רצה	4,671.67	43202	765,197.00
23	אם	4,444.59	43321	745,301.07
24	עם	4,333.17	43331	727,755.37
25	אמר	4,128.07	43196	681,096.31
26	אז	4,052.24	43202	653,014.96
27	סדר	4,305.52	42733	619,555.39
28	צריך	3,501.64	43101	554,553.56
29	רק	2,996.30	43306	492,899.21
30	חשב	3,021.85	43062	486,623.93

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
31	כאן	3,217.62	41759	482,525.32
32	הלך	3,297.97	43040	474,757.64
33	דבר	2,835.26	43192	460,896.72
34	איש	2,904.93	42958	447,039.01
35	אל	2,829.27	43249	445,302.82
36	כך	2,777.32	43151	439,802.19
37	יותר	2,682.46	43206	437,958.99
38	שם	2,640.94	43109	419,597.40
39	יכול	2,531.17	43141	395,488.52
40	ראה	2,399.17	43120	384,717.06
41	עכשיו	2,398.62	42758	376,752.89
42	אחד	2,308.83	43074	367,649.84
43	משהו	2,190.14	42768	347,937.78
44	למה	2,234.96	42608	347,496.06
45	בא	2,166.77	43050	337,851.62
46	זאת	2,365.90	41920	331,839.07
47	או	2,131.42	42796	327,755.51
48	זמן	2,054.03	43034	327,428.07
49	נכון	2,037.30	42700	316,377.56
50	כמו	2,002.99	42849	312,095.47
51	אין	1,945.44	42895	311,530.13
52	איך	1,898.80	42714	303,623.98
53	מי	1,927.41	42688	299,032.16
54	זו	2,012.55	38399	279,356.01
55	והיי	2,006.58	36676	267,355.76
56	כמה	1,691.00	42552	266,855.79
57	גם	1,657.26	42702	258,224.80
58	אולי	1,630.97	42239	248,128.64
59	נראה	1,603.47	42564	247,930.02
60	בית	1,689.04	40888	239,606.95
61	כדי	1,580.41	41152	235,870.54
62	קרה	1,541.01	42161	234,080.20

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
63	דיבר	1,495.64	41648	228,461.87
64	פעם	1,479.18	42191	227,908.16
65	דרך	1,431.59	41924	217,772.03
66	כ	1,396.49	42075	215,809.23
67	באמת	1,443.69	41591	213,274.80
68	הגיע	1,383.15	41984	212,188.29
69	מן	1,342.58	42071	208,680.69
70	חייב	1,399.35	40994	205,875.28
71	אחר	1,319.13	41924	204,371.79
72	עוד	1,343.73	42041	203,716.90
73	יום	1,374.21	41382	203,492.76
74	פשוט	1,427.29	40438	202,151.77
75	תודה	1,390.71	40779	201,382.26
76	כי	1,431.30	38980	200,396.45
77	כבר	1,292.23	41870	196,833.94
78	ילד	1,478.80	39003	196,063.68
79	אהב	1,378.72	40244	195,979.56
80	חיים	1,325.23	41514	195,630.26
81	בן	1,397.82	40029	189,652.26
82	מישהו	1,230.50	40919	185,007.30
83	קיבל	1,258.75	40776	182,920.92
84	מאוד	1,249.26	40437	180,116.99
85	לפני	1,165.26	41249	177,825.48
86	אלה	1,205.87	38074	175,864.37
87	אף	1,156.10	40829	174,348.57
88	עד	1,126.08	41190	172,739.78
89	הרבה	1,109.41	41188	169,322.17
90	רגע	1,138.53	40784	167,602.20
91	שנה	1,131.09	39679	163,922.25
92	עדיין	1,074.35	40811	162,899.29
93	עצמו	1,058.48	41000	161,896.89
94	האם	1,301.20	31767	160,232.98

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
95	ניסה	1,041.05	40669	157,586.42
96	חזר	1,047.71	40579	155,734.53
97	מצא	1,067.02	39632	152,592.88
98	מקום	1,022.76	40314	152,190.76
99	מת	1,080.04	39030	151,080.12
100	איפה	1,029.43	38203	147,222.19
101	אלוהים	1,088.19	35618	146,740.38
102	אדם	1,032.75	38089	142,442.27
103	הצטער	983.47	38552	141,426.15
104	עבר	935.03	40252	140,794.43
105	הכיל	947.86	34316	139,697.20
106	הבין	921.06	40099	138,979.22
107	חבר	964.33	38452	137,171.10
108	גדול	940.92	39208	136,557.19
109	איזה	924.60	39606	135,631.78
110	ממש	984.20	37369	135,556.18
111	בוא	946.82	38124	132,984.64
112	נתן	887.93	39452	132,919.69
113	קצת	904.88	38554	132,689.73
114	שמע	883.99	39499	132,389.87
115	עבודה	926.65	37349	131,496.42
116	הנה	911.68	38711	130,297.94
117	קדימה	1,026.32	34380	129,949.65
118	שני	871.35	39248	128,672.38
119	עזר	861.50	38806	125,101.16
120	יצא	838.60	39369	124,499.62
121	ובכן	974.64	27119	124,415.44
122	כש	819.06	38893	122,574.91
123	שוב	812.25	39393	121,475.41
124	לילה	873.88	35873	121,454.78
125	יד	840.78	37277	120,281.50
126	היום	831.50	37991	120,063.40

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
127	בדיוק	795.28	38931	118,899.34
128	אחת	795.49	39146	118,611.48
129	פה	924.87	32392	116,621.23
130	בבקשה	823.33	36882	116,224.18
131	הגיד	786.67	35355	113,058.11
132	אי	758.57	38291	111,108.95
133	קטן	767.54	37651	110,553.07
134	שום	755.65	36788	110,128.53
135	הרגיש	767.59	36977	108,848.61
136	אמא	862.25	26720	108,596.03
137	בטוח	731.46	38426	108,451.32
138	אפילו	721.87	38453	108,303.40
139	קשר	757.66	36266	107,386.86
140	קרא	722.63	37324	106,028.92
141	חדש	736.64	37387	105,906.04
142	תמיד	715.90	37943	105,650.60
143	אחרי	712.94	37831	105,300.02
144	אבא	820.11	29216	103,670.52
145	בשביל	732.76	35703	103,305.56
146	האמין	691.02	37080	100,444.06
147	בעיה	687.42	36380	99,810.93
148	הכיר	677.77	36335	99,235.94
149	התכוון	697.25	35138	98,459.83
150	סיפר	691.75	35660	97,955.32
151	מר	757.16	26570	95,214.18
152	שלום	694.08	34158	94,865.61
153	תן	653.40	35753	94,253.34
154	אה	813.58	22759	93,461.70
155	בטח	643.27	35618	92,057.76
156	כסף	701.11	28087	90,607.92
157	שעה	629.52	34939	90,469.46
158	עבד	615.84	35370	89,600.67

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
159	הביא	608.22	36660	89,451.10
160	מדי	605.77	34092	88,988.08
161	נמצא	624.10	34685	88,554.51
162	בגלל	627.50	33952	88,193.45
163	אחרון	597.58	36237	87,766.60
164	הרג	665.89	29925	87,503.76
165	ספר	650.41	32397	87,101.46
166	מוכן	595.09	35648	87,047.02
167	עניין	596.93	34716	86,003.11
168	לקח	566.91	35652	83,585.74
169	גרם	572.16	35168	83,370.70
170	גבר	616.60	31398	82,991.22
171	סיבה	587.32	34419	82,978.18
172	לב	578.66	34293	82,220.28
173	ראש	577.71	33647	81,881.27
174	אפשר	577.08	33563	81,809.19
175	שאל	548.02	34201	80,204.19
176	חברה	581.20	31444	79,229.20
177	עמד	532.42	34456	78,025.07
178	אכל	545.16	34354	77,965.88
179	חדר	562.65	31987	77,850.34
180	קשה	520.41	34923	76,722.92
181	אדוני	617.58	23260	76,519.32
182	התחיל	515.97	35015	76,507.15
183	רב	537.32	32606	75,953.11
184	הניח	518.51	33988	75,032.29
185	עולם	548.18	31680	75,017.87
186	נשאר	507.27	33880	73,774.10
187	תראה	517.75	31667	73,242.61
188	שב	492.98	34110	72,989.04
189	מקרה	506.88	32986	72,812.54
190	משפחה	539.47	29823	72,784.69

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
191	בוקר	511.84	30693	72,209.75
192	עזאזל	517.82	28613	69,812.27
193	כלום	480.33	31268	68,572.37
194	חוץ	468.18	32860	68,572.21
195	נכנס	466.72	33029	68,256.57
196	שבוע	475.15	30773	68,134.99
197	הו	560.98	17359	67,849.84
198	הכי	479.66	31255	67,765.96
199	אמור	462.96	33337	67,605.95
200	די	474.97	31259	67,154.39
201	חושב	479.44	28956	66,446.25
202	עסק	471.81	29969	66,443.09
203	חלק	453.04	32785	66,248.29
204	סוף	453.48	31625	65,679.31
205	בת	483.08	28383	65,290.77
206	ביותר	456.85	28965	64,134.84
207	עזב	438.85	31038	63,029.01
208	מצב	434.39	31396	62,470.12
209	זהו	456.73	28003	62,452.11
210	אינו	502.82	22002	62,360.35
211	שמר	419.67	31883	61,599.30
212	פנים	428.38	31491	61,479.32
213	בלי	425.60	31103	61,449.68
214	יפה	430.19	29667	61,276.12
215	חיפש	427.55	31018	61,268.16
216	הביתה	429.84	29329	60,905.53
217	עובד	417.59	30543	60,455.00
218	עבור	443.04	24695	60,153.44
219	בין	411.64	31121	59,891.76
220	רע	411.00	30810	59,707.99
221	הפך	412.11	30971	59,488.44
222	אמת	419.64	29720	59,396.20

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
223	כאילו	422.08	30119	59,394.22
224	אוקיי	499.98	11607	59,144.42
225	כמובן	410.97	29256	58,631.74
226	עיר	433.88	26088	57,827.31
227	הספיק	389.09	31940	57,607.73
228	אוכל	407.47	29281	57,359.32
229	מעולם	398.10	28830	57,351.62
230	השתמש	395.83	30709	57,198.31
231	שמח	400.63	30078	57,072.51
232	זכר	397.52	29885	56,908.95
233	המשיך	385.28	30977	56,696.83
234	דקה	394.64	28622	56,299.13
235	אמיתי	392.30	29258	56,195.03
236	העליי	399.75	24793	56,037.05
237	יחיד	380.58	31360	55,808.32
238	בעל	395.08	27783	55,099.53
239	נהדר	398.01	26285	54,808.89
240	אכפת	375.84	29977	54,622.15
241	קודם	369.46	31900	54,390.60
242	אלו	412.35	21603	54,342.83
243	תוכנית	398.15	26625	54,262.49
244	כדאי	381.19	28723	54,141.97
245	משחק	418.83	24065	53,994.07
246	חשוב	364.04	29954	53,320.25
247	ביקש	367.36	28968	53,226.56
248	נעשה	363.45	30093	53,192.10
249	נשמע	360.32	30269	52,698.89
250	מכונית	406.22	21273	52,631.30
251	לעולם	367.12	28630	52,465.45
252	מספר	375.71	25912	52,401.84
253	סליחה	375.07	25859	51,849.38
254	נחמד	361.55	27659	51,636.20

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
255	התקשר	366.95	25533	51,491.69
256	עין	355.23	28544	51,095.14
257	קיווה	339.39	29695	49,882.57
258	סיפור	359.30	26111	49,756.49
259	שאלה	346.90	27781	49,697.00
260	בחור	353.18	25592	49,550.29
261	חכה	353.50	25911	49,428.77
262	קרוב	334.26	29164	49,081.52
263	שינה	333.20	29600	48,874.52
264	הפסיק	332.59	28808	48,461.70
265	לעזאזל	354.18	24177	48,195.05
266	הודה	338.13	26896	48,159.38
267	כתב	352.64	23615	48,051.14
268	עלה	327.08	28020	48,007.14
269	מהר	336.87	27051	47,727.53
270	מוות	348.60	24487	47,473.71
271	אופן	327.33	26925	47,198.15
272	טלפון	344.31	22700	46,849.00
273	ישן	324.50	27460	46,580.60
274	תרא	329.43	25651	46,288.16
275	מחר	325.52	25269	46,192.82
276	לאן	319.10	27273	46,008.24
277	בכלל	314.78	27698	45,909.59
278	אך	385.52	15600	45,620.53
279	כוח	340.59	23354	45,303.74
280	רעיון	312.93	26575	45,051.22
281	לגבי	325.87	23852	45,041.46
282	ילך	305.77	27026	44,598.31
283	עצר	313.61	26180	44,452.62
284	מוזר	313.45	25833	44,201.31
285	ללא	318.60	24590	44,127.48
286	מזל	305.24	26640	44,048.49

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
287	הצליח	305.24	25896	43,439.64
288	שנייה	302.19	26144	43,406.72
289	צדק	298.74	27334	43,322.78
290	גברת	331.04	20248	43,306.13
291	חיכה	295.03	27476	43,261.38
292	נוסף	303.43	25572	43,229.02
293	דלת	312.74	24076	43,186.33
294	אח	321.28	22129	43,183.74
295	חזרה	303.50	25849	43,145.96
296	חודש	301.43	24849	43,056.36
297	מתי	293.94	26975	42,811.45
298	חזק	298.58	25977	42,649.30
299	משטרה	323.01	18340	42,102.91
300	במקום	284.12	27901	42,077.75
301	סוג	294.47	24733	41,865.95
302	שיחק	302.95	23577	41,797.36
303	למד	294.15	24951	41,691.19
304	שלח	291.17	25495	41,631.27
305	חץ	288.29	25677	41,610.97
306	אחי	329.83	17859	41,573.60
307	דם	320.20	20297	41,477.92
308	חלה	310.72	21244	41,434.43
309	כמעט	281.40	27109	41,434.30
310	צוות	310.61	21080	41,346.42
311	ברור	286.40	25902	41,332.36
312	ערב	296.13	23196	41,310.28
313	וה	279.53	26863	41,281.91
314	דולר	314.65	18338	41,161.75
315	בחר	282.89	26160	41,120.38
316	חי	284.29	25663	40,760.78
317	כלל	278.64	25929	40,627.30
318	החזיק	282.37	25545	40,626.24

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
319	בדק	285.13	24773	40,478.32
320	לאחר	287.91	23190	40,342.26
321	כנראה	280.36	24952	40,073.26
322	כדור	303.73	20345	40,010.80
323	רוח	288.02	23413	39,800.36
324	הבחור	286.50	22331	39,689.71
325	מאוחר	269.37	25779	39,409.18
326	השאיר	268.59	26619	39,389.70
327	קנה	277.93	23964	39,203.17
328	רצח	310.41	16162	39,051.82
329	הוציא	266.24	25538	38,662.52
330	איתך	262.46	25409	38,471.02
331	מבין	260.84	25479	38,133.04
332	סיים	257.89	25437	37,552.96
333	התראה	266.97	22837	37,500.68
334	פחד	267.35	22795	37,440.14
335	שלוש	263.61	22791	37,439.85
336	למעשה	264.72	23015	37,394.30
337	משרד	277.96	19641	37,341.92
338	ככה	261.25	23796	37,325.42
339	שילם	265.52	22382	37,304.84
340	כאשר	289.88	15772	37,203.37
341	גרוע	254.92	25267	36,893.79
342	כבוד	266.43	21817	36,857.39
343	הבטיח	255.29	24500	36,843.15
344	חסר	252.86	24559	36,622.96
345	תמונה	268.53	20845	36,579.12
346	מלא	248.01	25189	36,390.04
347	לכן	257.71	22825	36,356.69
348	לבד	253.05	24642	36,334.20
349	שוטר	290.24	14924	36,263.50
350	איבד	251.51	24396	36,235.05

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
351	נסע	264.57	20354	36,215.36
352	השיג	249.66	23811	35,891.45
353	לגמרי	251.39	23158	35,749.77
354	החוצה	252.68	22603	35,733.34
355	לפחות	241.21	25727	35,549.87
356	נ	242.49	24902	35,447.84
357	במשך	244.12	23392	35,168.03
358	פרק	246.34	27236	35,138.22
359	איתי	241.34	24239	35,115.36
360	חושבת	248.37	22479	34,933.82
361	פגע	245.53	23466	34,885.83
362	הת	238.73	24614	34,765.13
363	בחייך	250.63	20788	34,647.91
364	סרט	270.08	16423	34,209.10
365	שכח	234.51	23959	34,131.92
366	בבקש	236.31	23109	34,115.62
367	צעיר	242.48	21498	34,054.05
368	ישב	233.75	23369	33,963.80
369	בהחלט	236.73	22666	33,917.54
370	שונה	235.15	23096	33,823.17
371	קח	237.86	21743	33,700.21
372	א	246.98	20598	33,669.57
373	צריכה	242.19	21649	33,561.11
374	מעל	233.18	22991	33,324.40
375	קל	227.86	24171	33,263.27
376	מטה	240.09	20269	33,183.99
377	ותק	249.98	18007	33,124.56
378	לשם	225.19	24044	33,098.96
379	אהבה	249.54	18292	33,084.77
380	יחד	234.73	21633	33,047.58
381	קורה	230.18	23024	32,834.36
382	הקשיב	228.03	22621	32,779.88

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
383	אתמול	234.91	21452	32,769.90
384	מילה	229.60	22043	32,737.27
385	נקודה	231.75	21657	32,715.73
386	הכול	269.42	11674	32,670.68
387	צורה	229.16	22438	32,619.72
388	נגע	235.17	21069	32,512.80
389	בלתי	227.90	21790	32,316.19
390	מים	241.56	18838	32,195.27
391	למעלה	229.19	20843	32,169.08
392	מושג	222.77	23299	32,167.89
393	פתח	224.93	22423	32,150.08
394	נהג	228.98	20845	32,114.59
395	סתם	226.73	21609	32,045.04
396	היכן	249.05	16013	32,042.37
397	סלח	226.18	21207	31,969.96
398	הסתכל	221.65	22239	31,936.39
399	בתוך	224.12	21982	31,825.03
400	כוונה	219.25	23367	31,815.50
401	מייד	221.99	21960	31,754.50
402	מערכת	236.45	18981	31,750.62
403	נגמר	221.22	22750	31,673.90
404	הזדמנות	220.09	22444	31,575.73
405	תינוק	253.53	14886	31,544.58
406	הראה	216.72	22842	31,463.59
407	הערב	232.61	18671	31,374.79
408	עזרה	218.48	23031	31,327.98
409	אלא	215.41	22893	31,217.37
410	אתן	220.47	21524	31,029.50
411	סיכוי	216.52	22067	30,832.27
412	הפעם	211.44	23297	30,806.02
413	ניצח	225.98	19665	30,796.91
414	הציל	227.24	19759	30,788.14

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
415	נשק	231.36	18112	30,773.61
416	רופא	234.77	16659	30,569.07
417	שלושה	216.13	20973	30,552.98
418	י	217.36	20366	30,142.76
419	קול	224.59	17953	29,974.60
420	אוי	231.73	16449	29,966.62
421	כאב	216.83	19849	29,820.81
422	שתי	206.35	21679	29,695.86
423	אעשה	203.70	22636	29,539.90
424	כפי	210.91	19918	29,424.98
425	רציני	204.36	21409	29,310.72
426	הציע	205.25	21157	29,234.52
427	וואו	227.03	15605	29,183.80
428	כלא	224.46	15589	29,175.10
429	אדיר	218.21	17816	29,119.99
430	כלומר	219.69	16988	29,060.04
431	דין	226.93	14317	28,825.21
432	ביחד	208.36	19338	28,684.00
433	בעוד	200.88	21239	28,641.30
434	כרגע	205.01	19976	28,513.72
435	שיר	220.27	15685	28,468.29
436	מלחמה	222.93	14580	28,291.00
437	דעה	198.70	20700	28,162.33
438	כלב	222.64	14561	28,161.94
439	לפעמים	197.11	21002	28,138.62
440	כעת	223.23	14172	28,112.22
441	נעלם	202.86	19993	28,095.16
442	שיחה	200.39	19897	28,067.30
443	למען	200.50	19906	28,020.53
444	חמש	200.75	19160	27,981.72
445	רחוב	206.14	17507	27,813.91
446	נורא	198.72	19735	27,792.23

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
447	שניים	193.25	20788	27,652.46
448	מיוחד	196.17	20056	27,599.23
449	האליי	199.52	18630	27,555.89
450	ירד	192.70	20241	27,389.68
451	ודה	190.54	21077	27,347.06
452	קבוצה	209.25	15813	27,177.08
453	שאר	190.62	20718	27,160.26
454	זונה	207.61	15443	27,136.14
455	שכן	190.95	21089	27,135.19
456	נגד	196.68	18684	27,078.49
457	אלי	191.31	19944	26,985.67
458	יצר	197.48	18486	26,925.73
459	יופי	199.93	18075	26,916.58
460	ארץ	206.49	15669	26,750.94
461	מדינה	201.80	15768	26,725.20
462	תפס	189.40	20032	26,709.99
463	חוק	198.36	17001	26,665.18
464	גר	194.35	17985	26,457.55
465	החזיר	185.84	20104	26,053.44
466	גש	183.49	19488	25,872.35
467	אקדח	206.18	12963	25,829.10
468	שה	179.04	20798	25,746.59
469	מידע	194.28	16139	25,694.72
470	טיפל	182.77	19710	25,625.25
471	משפט	202.09	13243	25,502.57
472	גנב	189.43	17273	25,454.84
473	מסוגל	183.37	18869	25,425.40
474	תורגם	183.74	19872	25,182.90
475	ארוחה	182.99	18292	25,096.14
476	שקט	180.46	18622	25,074.92
477	צד	178.71	19226	24,993.82
478	אש	192.82	15071	24,903.81

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
479	מצטער	177.03	19572	24,824.49
480	אב	187.02	16121	24,752.28
481	ליד	173.76	20052	24,667.90
482	טעות	175.47	19445	24,620.91
483	פחות	172.52	19982	24,539.97
484	רגיל	173.06	19572	24,453.20
485	תיק	189.62	14271	24,444.04
486	גבוה	173.98	18756	24,392.46
487	מלך	207.75	9924	24,234.01
488	מדוע	192.68	12801	24,156.73
489	ניתן	171.44	18888	23,988.53
490	הגן	175.92	17830	23,933.98
491	הצלחה	169.09	19333	23,899.78
492	מספיק	167.51	20251	23,880.50
493	רכב	182.51	15119	23,842.73
494	כיוון	173.34	17992	23,836.28
495	פשע	183.46	14565	23,830.07
496	הורה	178.63	16310	23,742.47
497	הסכים	169.25	18770	23,706.65
498	הוריד	169.55	18273	23,499.72
499	לחץ	172.52	17264	23,480.23
500	דאג	169.20	18349	23,357.49
501	יכולת	166.59	19032	23,302.07
502	נפלא	174.49	16076	23,298.72
503	תדאג	164.66	19026	23,015.10
504	תחת	164.84	17993	22,927.89
505	הכין	165.99	18648	22,920.72
506	עץ	176.20	14898	22,901.55
507	הודעה	169.29	16692	22,803.00
508	חרא	184.92	11147	22,743.92
509	מוח	172.45	15815	22,731.49
510	מטרה	170.87	15951	22,708.94

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
511	גוף	170.08	16491	22,704.01
512	הא	178.76	12766	22,467.57
513	אצל	162.63	17589	22,416.64
514	מצחיק	163.76	16994	22,215.93
515	שנא	162.46	17318	22,179.80
516	לפ	160.76	17418	22,154.31
517	בגד	161.62	17232	22,057.70
518	סימן	162.60	16983	22,005.76
519	שווה	157.99	18112	21,964.90
520	קטע	164.09	16392	21,940.73
521	דוד	177.08	12007	21,892.81
522	עלול	158.56	17521	21,732.89
523	רוב	158.26	17127	21,717.56
524	כוכב	174.76	12372	21,684.59
525	העביר	155.94	17975	21,654.58
526	אפשרי	156.08	17658	21,533.04
527	פגישה	163.23	15003	21,525.77
528	אור	159.84	16257	21,492.34
529	מין	160.27	15961	21,412.36
530	ביי	167.40	12971	21,361.19
531	מנהל	163.33	14360	21,346.57
532	בנה	155.16	17246	21,258.71
533	ארוך	151.52	18330	21,245.24
534	זוכר	156.21	16942	21,175.54
535	תפקיד	158.07	15537	21,088.01
536	נעל	157.99	15835	21,039.47
537	ציפה	149.55	18426	20,981.28
538	מוקדם	149.54	18294	20,918.57
539	בקרוב	150.88	18039	20,845.83
540	מתוק	158.32	15261	20,818.09
541	הסביר	149.51	17836	20,770.69
542	אסור	152.51	16451	20,551.71

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
543	איתן	151.03	16540	20,453.49
544	לבש	151.09	16629	20,447.32
545	לפי	150.84	16438	20,426.84
546	מעבר	148.95	16944	20,415.67
547	דירה	160.75	13077	20,404.80
548	סם	166.80	10829	20,336.70
549	רחוק	147.68	17301	20,331.26
550	שתיים	151.05	15645	20,273.31
551	ניו	163.64	11132	20,272.95
552	בדיקה	158.91	13232	20,266.49
553	מאשר	145.52	17633	20,242.52
554	זוג	152.22	15370	20,178.23
555	עובדה	145.57	17294	20,159.08
556	הופיע	146.45	16743	20,067.84
557	אוויר	151.69	14943	20,004.89
558	החלטה	148.45	15978	20,004.06
559	זז	148.03	16105	19,943.30
560	גידי	149.70	15672	19,940.37
561	מעט	146.86	16294	19,918.81
562	כרטיס	153.30	13967	19,915.97
563	טיפש	147.63	16103	19,894.76
564	מפה	152.42	14189	19,813.95
565	שירות	148.24	15767	19,803.99
566	אישי	143.37	16623	19,654.34
567	ערך	143.65	16806	19,629.98
568	קיים	144.55	16353	19,591.61
569	שחור	151.78	13552	19,568.42
570	עורך	154.40	12270	19,564.62
571	זקוק	146.62	15699	19,544.02
572	בחורה	149.06	14657	19,543.95
573	התמודד	143.53	16712	19,517.27
574	נלחם	150.51	13749	19,341.28

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
575	מיטה	145.10	15212	19,256.18
576	הזמין	141.84	16304	19,217.46
577	מתחת	140.67	16600	19,147.46
578	מחדש	141.67	16186	19,137.04
579	אלך	139.13	17337	19,077.20
580	מפקד	169.23	6732	19,031.79
581	אימא	170.32	6297	18,948.91
582	המ	143.47	14494	18,834.71
583	הלו	145.96	13386	18,822.25
584	משך	137.63	16544	18,810.59
585	מהלך	140.43	15503	18,797.20
586	בערך	137.83	16236	18,769.03
587	חומר	143.05	14014	18,692.30
588	אית	135.28	17189	18,673.98
589	חלום	146.78	12785	18,651.13
590	שחרר	139.94	15146	18,509.63
591	בתור	137.75	15608	18,417.04
592	ברח	138.25	15514	18,392.03
593	שולחן	138.30	14963	18,390.26
594	הוטרף	136.60	15580	18,325.60
595	נפגש	134.66	16054	18,303.86
596	למרות	135.05	16043	18,296.82
597	צעד	136.83	15122	18,273.83
598	צוחק	140.05	13873	18,252.69
599	קפה	141.11	13261	18,171.09
600	שאמר	131.03	17308	18,158.57
601	מאחורי	132.70	16436	18,117.82
602	יחסים	137.74	13752	17,848.19
603	גב	139.40	12727	17,799.65
604	חג	154.76	7682	17,735.78
605	מס	144.25	10714	17,697.20
606	חדשות	132.52	15380	17,684.76

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
607	לבן	134.60	14048	17,672.25
608	נרגע	134.64	14117	17,651.29
609	ספק	130.46	15635	17,642.46
610	מושלם	131.33	15526	17,632.35
611	צהריים	131.93	14887	17,598.78
612	רשימה	136.18	13403	17,594.75
613	גמור	130.97	14927	17,456.99
614	יורק	143.43	9829	17,434.56
615	חשבון	131.59	14268	17,373.51
616	זכות	131.24	14256	17,366.56
617	שר	142.37	10350	17,362.96
618	ארבע	130.65	14298	17,328.25
619	התאים	127.22	16114	17,305.27
620	עלייך	131.93	14499	17,246.79
621	חם	129.91	14682	17,242.57
622	שלומך	130.64	13675	17,222.87
623	עתיד	133.23	13385	17,180.78
624	נפל	127.96	15183	17,180.54
625	ים	138.23	11409	17,121.31
626	הכניס	125.56	15835	17,079.77
627	ברוך	128.21	14984	17,074.61
628	טעם	127.69	15311	17,071.57
629	כיף	131.81	14072	17,065.65
630	נשיא	151.43	6364	17,038.40
631	תא	134.30	12524	16,978.58
632	סביבה	126.17	15259	16,969.43
633	נהנה	126.79	15334	16,940.34
634	חמוד	130.86	13620	16,890.01
635	רצינות	126.73	15123	16,885.75
636	קו	130.14	13233	16,876.31
637	קפטן	151.94	6009	16,848.27
638	תחנה	132.88	12378	16,835.83

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
639	מיליון	134.79	10942	16,820.80
640	תקשיב	127.70	14245	16,801.39
641	זקן	130.64	12838	16,747.65
642	הוביל	125.32	14940	16,687.58
643	מאה	126.35	14247	16,677.42
644	אמצע	122.43	15917	16,665.68
645	זכה	128.73	13316	16,657.83
646	משימה	135.30	11277	16,655.25
647	מותק	132.18	12259	16,611.76
648	סמך	125.34	14787	16,598.62
649	מטוס	140.52	8624	16,593.01
650	מיני	125.26	14421	16,494.92
651	אזור	127.80	13129	16,439.94
652	פנימה	123.61	14506	16,415.87
653	חנות	129.09	12676	16,402.96
654	פעולה	124.84	13885	16,366.53
655	שטח	127.03	13205	16,338.87
656	הרגל	123.80	14088	16,306.13
657	סבל	122.28	14809	16,303.65
658	תשובה	121.96	14878	16,263.69
659	אקח	119.79	15773	16,225.58
660	עשר	123.75	13697	16,197.07
661	תפסיק	123.12	13940	16,126.21
662	הזכיר	118.09	15934	16,081.89
663	נשא	121.55	13857	15,973.28
664	ן	118.91	15104	15,938.03
665	נקרא	118.08	15250	15,882.33
666	אבי	130.11	10450	15,857.54
667	מכר	121.68	13216	15,842.08
668	ראייה	126.22	11606	15,827.21
669	צא	122.33	13087	15,785.69
670	לתוך	119.92	14037	15,756.47

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
671	כה	123.13	12452	15,657.72
672	הצטרך	116.10	15287	15,646.91
673	החליט	116.23	14991	15,612.97
674	ביצע	119.32	13684	15,599.03
675	בניין	125.13	11215	15,544.10
676	מול	117.33	14323	15,530.03
677	קצר	115.08	15089	15,517.74
678	מחשב	129.25	9644	15,504.76
679	נעים	117.80	13761	15,494.13
680	במיוחד	114.45	15147	15,460.49
681	מלון	126.80	9865	15,419.08
682	המון	118.28	13256	15,360.30
683	אדום	120.91	11969	15,329.05
684	שן	115.04	15141	15,327.05
685	מוצא	114.63	14629	15,319.49
686	שייך	116.56	14040	15,313.17
687	הבחורה	120.11	12398	15,280.28
688	בצד	114.58	14506	15,262.39
689	ניסיון	114.03	14770	15,245.69
690	תרופה	126.29	10022	15,244.60
691	חקירה	125.22	10087	15,243.61
692	שש	117.74	12825	15,190.30
693	עשוי	117.16	13352	15,169.71
694	יחידה	119.76	12092	15,144.45
695	החליף	113.59	14641	15,130.44
696	התחלה	112.05	15161	15,125.40
697	גילה	113.89	14658	15,121.56
698	תוך	114.03	14170	15,109.20
699	חתיכה	115.55	13661	15,101.00
700	אתר	119.61	12246	15,089.12
701	נוח	113.84	14384	15,081.07
702	חתך	115.55	13556	15,078.73

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
703	צבע	118.29	12078	15,038.51
704	מצוין	117.77	12431	15,035.63
705	צורך	112.41	14581	14,963.92
706	אדון	124.45	9232	14,930.77
707	שער	119.87	11155	14,928.20
708	יקר	112.60	14095	14,892.74
709	העדיף	111.09	14882	14,882.73
710	ככל	111.53	14377	14,824.95
711	מסוכן	113.11	13987	14,810.44
712	חבל	112.44	13954	14,784.34
713	הגנה	116.55	12072	14,779.16
714	גיל	114.61	12759	14,774.09
715	הצטרף	111.71	14144	14,726.63
716	ישר	110.98	13788	14,712.58
717	שינוי	112.46	13717	14,706.12
718	דובר	114.53	12523	14,660.95
719	אראה	109.54	14588	14,608.22
720	הרס	111.16	14270	14,601.03
721	שותף	116.14	11556	14,592.88
722	תהי	111.41	13941	14,582.70
723	לכי	115.25	12188	14,577.24
724	אדמה	117.03	11340	14,570.31
725	ענה	109.98	14788	14,536.55
726	אחראי	111.20	13581	14,514.11
727	מסר	111.96	13107	14,511.56
728	קורבן	123.17	8406	14,461.35
729	סכנה	111.46	13566	14,457.71
730	פיטר	127.62	6788	14,451.25
731	הרשה	108.65	14105	14,425.88
732	זרק	110.00	13618	14,398.68
733	שיעור	114.08	11679	14,339.70
734	הדבר	108.74	14108	14,327.52

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
735	אידיוט	112.21	12174	14,296.85
736	מפני	117.21	10575	14,277.73
737	שליטה	109.77	13108	14,205.55
738	עונה	109.04	16314	14,204.99
739	אשר	116.53	9766	14,180.13
740	כלשהו	109.14	13207	14,169.95
741	אגיד	108.06	13702	14,149.72
742	סגר	107.96	13586	14,147.49
743	גדל	107.92	13562	14,142.58
744	מבט	107.36	13651	14,119.98
745	צפה	107.34	14047	14,115.97
746	הדה	108.86	13101	14,100.96
747	ספינה	127.94	5607	14,081.82
748	ניתוח	121.60	7930	14,079.55
749	אלף	114.09	10277	14,065.09
750	מהיר	107.88	13155	14,017.46
751	רמה	106.85	12815	13,822.44
752	תוצאה	107.69	12297	13,775.40
753	חכם	104.96	13425	13,732.00
754	פרטי	105.91	12665	13,679.34
755	השתנה	105.04	13421	13,674.09
756	מתנה	107.54	12112	13,647.45
757	ירה	109.72	10718	13,616.51
758	הפריע	103.19	13961	13,612.51
759	טיפול	109.18	10912	13,513.34
760	אמריקני	111.85	9337	13,508.98
761	שקר	105.60	12361	13,480.02
762	נושא	104.50	12484	13,473.45
763	מחלקה	108.70	10556	13,439.33
764	עוזב	102.84	13289	13,398.49
765	שמחה	103.53	13077	13,386.36
766	חייל	115.20	7859	13,371.59

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
767	מייקל	122.72	5053	13,346.81
768	ניהל	102.32	12930	13,344.14
769	חשבתי	100.66	14490	13,331.88
770	עקב	103.58	12874	13,327.30
771	הסתובב	101.50	13377	13,315.83
772	איי	116.05	7139	13,300.25
773	חופשי	102.71	12818	13,274.52
774	כלי	105.06	11784	13,260.23
775	צבא	111.92	8648	13,250.48
776	מועדון	110.87	8972	13,243.52
777	גמר	102.83	12307	13,210.04
778	מחיר	103.97	11950	13,204.80
779	הלוואה	101.61	13220	13,171.05
780	היקח	100.47	13467	13,160.30
781	ביטחון	103.94	11698	13,131.53
782	פעל	101.72	12705	13,110.08
783	הצעה	103.81	11235	13,030.21
784	לקוח	107.94	9252	12,987.38
785	תאונה	105.63	10322	12,944.52
786	מפתח	104.54	10730	12,935.46
787	לישון	100.94	12295	12,904.31
788	הגיוני	99.27	13055	12,859.74
789	מתוך	98.82	12809	12,816.10
790	לחלוטין	99.56	12506	12,798.11
791	אפשרות	98.89	12787	12,787.30
792	ודאי	104.29	9844	12,661.30
793	שחקן	105.82	8651	12,613.95
794	סוכן	106.83	8570	12,590.35
795	תני	98.64	12338	12,578.92
796	רץ	98.53	11630	12,532.24
797	ההוא	99.43	11584	12,522.00
798	שלם	95.55	13358	12,492.23

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
799	שלב	98.98	11500	12,478.43
800	פתוח	96.43	12827	12,465.94
801	חלון	98.73	11414	12,448.83
802	איתה	96.34	12929	12,412.61
803	עמוק	96.44	12460	12,393.48
804	מרכז	99.29	10929	12,382.10
805	שומר	97.66	11831	12,352.66
806	פ	97.73	11502	12,326.17
807	מעמד	96.65	11840	12,313.84
808	שעשה	93.56	13540	12,283.06
809	חשש	96.14	12296	12,279.36
810	ק	104.35	8246	12,276.94
811	חך	97.29	11749	12,275.74
812	הוכיח	96.36	11951	12,263.14
813	ייתכן	100.63	10136	12,259.43
814	שנוכל	94.29	13339	12,227.21
815	טום	109.65	5829	12,227.18
816	אצטרך	93.98	13342	12,208.73
817	תקופה	95.31	11999	12,165.49
818	כבד	95.19	11980	12,159.65
819	מהירות	97.15	10975	12,149.82
820	שכר	96.45	11152	12,141.03
821	שלט	95.73	12031	12,131.46
822	התחתן	100.46	9118	12,116.66
823	לפה	99.95	9858	12,108.53
824	יכל	95.77	11541	12,093.68
825	מסוים	94.24	12276	12,081.97
826	מרחק	95.23	11900	12,074.07
827	מאושר	96.31	11038	12,061.54
828	החלק	93.62	12666	12,044.81
829	בעצם	95.39	11350	12,038.70
830	בירה	97.97	10361	12,027.57

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
831	חייך	93.32	12577	12,001.40
832	בר	95.34	11298	11,991.55
833	עוזר	94.92	11674	11,975.83
834	רגל	94.51	11455	11,975.62
835	בנק	102.96	7262	11,942.09
836	העריך	92.03	12680	11,909.84
837	זיהה	93.99	11751	11,892.34
838	טלוויזיה	98.20	9566	11,874.32
839	כביש	97.48	9677	11,833.28
840	האשים	92.69	12138	11,829.36
841	תגיד	91.94	12195	11,809.98
842	אכן	95.50	10630	11,809.60
843	נצטרך	92.24	12496	11,803.66
844	וב	91.07	12447	11,799.22
845	שמונה	93.89	11034	11,794.78
846	פרנק	109.13	4156	11,722.69
847	פרט	91.84	11925	11,704.02
848	נישואין	97.21	9134	11,654.43
849	שדה	95.58	9777	11,653.74
850	אביך	97.35	9280	11,646.50
851	ם	93.39	10857	11,622.68
852	עצור	94.74	9969	11,598.82
853	נפגע	91.83	11568	11,584.18
854	ידיד	93.52	10365	11,524.33
855	קרב	93.78	10203	11,497.44
856	ר	93.64	10008	11,482.76
857	קלט	92.53	10591	11,441.88
858	תסתכל	90.70	11244	11,381.56
859	מכתב	96.48	7863	11,380.72
860	הפחיד	89.45	11866	11,351.92
861	השנה	91.50	10483	11,348.33
862	שכנע	89.20	11976	11,345.38

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
863	התרחק	90.19	11527	11,334.51
864	רגש	91.16	10886	11,309.03
865	שבר	89.28	11673	11,304.57
866	התקרב	88.58	12090	11,294.13
867	מעניין	88.40	12116	11,288.93
868	גישה	90.20	11205	11,286.08
869	הוגן	89.12	11626	11,273.59
870	ע	91.50	10061	11,245.14
871	הללו	95.52	8080	11,228.36
872	ויתר	88.70	11732	11,217.31
873	קר	89.51	10900	11,167.88
874	שופט	95.67	7427	11,148.90
875	נפטר	88.40	11360	11,146.73
876	צפון	91.14	9785	11,106.91
877	עדיף	86.97	11850	11,039.94
878	אירוע	88.79	10547	11,012.65
879	נו	90.12	10079	11,006.79
880	ברית	92.53	8500	10,977.05
881	הבחר	87.17	11356	10,964.78
882	ארבעה	88.03	10579	10,964.41
883	סמל	95.67	6868	10,940.66
884	רעב	88.17	10755	10,937.00
885	רצון	87.76	10723	10,908.17
886	דן	92.91	7700	10,881.90
887	הכה	87.37	10523	10,853.90
888	אבן	92.09	8200	10,849.61
889	איום	87.46	10677	10,847.59
890	פגש	85.17	11762	10,845.77
891	זבל	89.73	9105	10,803.07
892	הידי	85.94	11046	10,770.86
893	הסתיים	85.13	11512	10,742.17
894	הלאה	84.86	11215	10,654.23

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
895	נקי	85.41	10838	10,649.69
896	לאט	87.63	9232	10,607.85
897	סקס	91.36	7981	10,597.71
898	לאחרונה	83.74	11795	10,589.28
899	אחרת	82.17	12093	10,543.32
900	תראו	84.65	10783	10,465.97
901	זהיר	83.93	10768	10,458.12
902	זין	92.29	6072	10,437.30
903	התגעגע	84.83	10209	10,406.50
904	תקווה	83.91	10662	10,396.82
905	אבטחה	88.36	8487	10,391.76
906	חטף	85.14	9866	10,383.19
907	ראוי	83.35	10716	10,379.05
908	כעס	83.58	10766	10,363.10
909	נחש	82.60	11043	10,307.23
910	תכנן	81.84	11352	10,289.05
911	גיבור	86.81	8501	10,276.52
912	מולד	92.44	5476	10,266.64
913	פרס	87.24	7932	10,264.43
914	מכירה	84.74	9343	10,253.83
915	אנושי	86.22	9023	10,252.97
916	ג	87.19	8009	10,233.42
917	היסטוריה	82.86	10043	10,210.95
918	שהייה	80.11	11779	10,198.62
919	עבורך	83.65	9891	10,193.19
920	מחשבה	81.10	11067	10,161.03
921	סביב	81.09	10810	10,149.76
922	סגן	90.44	5969	10,149.25
923	פנה	81.00	10759	10,123.09
924	התעורר	81.45	10636	10,092.96
925	הריח	82.25	10142	10,056.82
926	תיקן	81.78	10546	10,053.67

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
927	עלי	79.63	11528	10,053.16
928	גבול	82.24	9761	10,015.48
929	רשת	83.90	8897	10,008.24
930	נשבע	80.59	10431	9,994.46
931	אמריקה	84.61	8024	9,991.76
932	יגע	80.51	10380	9,978.10
933	מידה	79.88	10674	9,973.10
934	זהב	85.44	7341	9,941.34
935	כיצד	84.64	7981	9,929.39
936	מישהי	80.92	10164	9,872.38
937	נדבר	78.53	11122	9,865.93
938	כניסה	79.45	10563	9,865.11
939	מקור	81.02	9702	9,859.33
940	מעשה	79.71	10331	9,846.09
941	ממשלה	85.59	6709	9,814.41
942	היטב	79.21	10445	9,808.53
943	תיכון	82.49	8548	9,793.41
944	מנה	80.37	9390	9,756.09
945	ביקר	78.52	10469	9,745.40
946	אורח	79.08	10094	9,730.11
947	עשרה	80.04	9185	9,712.40
948	נהרג	80.13	9546	9,700.47
949	חתונה	87.07	5747	9,697.32
950	הדע	77.32	11125	9,695.32
951	ריק	81.76	8480	9,686.53
952	דרש	77.86	10705	9,674.68
953	בפני	77.94	10488	9,658.03
954	מורה	82.34	7817	9,626.10
955	הפה	78.57	9761	9,620.86
956	צחק	78.02	10313	9,614.25
957	מוכר	77.12	10646	9,599.85
958	שטות	77.72	9762	9,593.07

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
959	תלוי	76.01	11185	9,576.28
960	החבב	79.58	9143	9,560.52
961	הושלם	77.06	10624	9,552.39
962	קבע	76.58	10636	9,546.81
963	רואה	77.06	10948	9,544.91
964	משום	80.34	8509	9,535.93
965	גן	79.64	8807	9,526.59
966	טיסה	82.21	7279	9,523.55
967	אלייך	78.14	9960	9,517.83
968	עשי	76.60	10993	9,511.87
969	מקומי	77.49	9921	9,504.75
970	הפסיד	77.96	9441	9,500.86
971	סגור	76.32	10525	9,483.09
972	חן	78.86	8825	9,469.23
973	שלישי	76.90	9673	9,433.39
974	חירום	79.14	8933	9,419.71
975	גאה	76.32	10180	9,417.79
976	התחה	77.55	9242	9,416.76
977	תנועה	77.32	9386	9,413.11
978	לשעבר	77.90	9252	9,412.99
979	טען	77.16	9634	9,408.22
980	פול	85.39	5004	9,406.32
981	ארון	79.30	8585	9,390.43
982	אגב	75.36	10783	9,367.74
983	נערה	79.83	7758	9,339.59
984	נער	78.44	8471	9,336.01
985	העמיד	75.05	10781	9,329.74
986	בעצמך	73.90	11272	9,306.95
987	השקר	76.21	9914	9,301.44
988	חש	76.70	9585	9,300.91
989	עצמך	73.87	11145	9,294.70
990	הסתדר	74.51	10726	9,283.55

RANK	LEMMA	FREQUENCY	RANGE	U_{DP}
991	תת	75.21	10301	9,279.69
992	תחושה	75.66	9992	9,278.80
993	תקף	77.29	9368	9,266.38
994	סאם	88.78	2995	9,254.07
995	עשית	73.74	11109	9,239.13
996	בסיס	78.21	8149	9,234.94
997	ראשי	75.85	9411	9,231.71
998	דיווח	76.23	9390	9,221.82
999	המשך	74.32	10453	9,211.77
1000	בוודאי	77.55	8438	9,210.24

Appendix B: Scripts

Appendix B.1: HebrewLemmaCount.py

```
#! /usr/bin/env python3
# -*- coding: utf-8 -*-

import re
import os
import gzip
from collections import defaultdict


############################################################
# ----------------- INITIALIZE VARIABLES ----------------- #
############################################################

# Define path for topmost directory to search. Make sure this points to
# the correct location of your corpus.
corpus_path = './OpenSubtitles2018_parsed_single'

# Initialize dictionaries
lemma_by_corpus_dict = {}
lemma_totals_dict = {}
token_count_dict = {}
lemma_DPs_dict = defaultdict(float)
lemma_UDPs_dict = defaultdict(float)

total_tokens_int = 0
table_list = []

```

```
# Set size of final list
list_size_int = 5000


#################################################################
# ------------------- DEFINE FUNCTIONS ------------------- #
#################################################################


# Open XML file and read it.
def open_and_read(file_loc):
    with gzip.open(file_loc, 'rt', encoding='utf-8') as f:
        read_data = f.read()
    return read_data


# Search for lemma and add counts to "frequency{}".
def find_and_count(doc):
    corpus = str(f)[38:-4]
    match_pattern = re.findall(r'lemma="[א-ת]+"', doc)
    for word in match_pattern:
        if word[7:-1] in lemma_by_corpus_dict:
            count = lemma_by_corpus_dict[word[7:-1]].get(corpus, 0)
            lemma_by_corpus_dict[word[7:-1]][corpus] = count + 1
        else:
            lemma_by_corpus_dict[word[7:-1]] = {}
            lemma_by_corpus_dict[word[7:-1]][corpus] = 1


#################################################################
# ------------------- OPEN AND READ ------------------- #
#################################################################

```

```
# Open and read all files. If calculating only for a specific
↪ language,
# comment out this code and uncomment the large block that follows.
#
for dirName, subdirList, fileList in os.walk(corpus_path):
    if len(fileList) > 0:
        f = dirName + '/' + fileList[0]
        find_and_count(open_and_read(f))

#################################################################
# ---------------- LANGUAGE-SPECIFIC BLOCK -----------------
#
# This large block of code is for creating a list using only movies
↪ #
# with a specific primary language (in this case, Hebrew). Be sure
↪ to #
# uncomment the relevant lines of code, and to comment out the block
↪ #
# above. #
#
#
# Create list of IDs for movies with Hebrew as primary language. #
# This makes use of a text file that must already exist with this
↪ list. #
#
# Hebrew_IDs_list = []
# with open('./Hebrew_originals.txt', 'r', encoding='utf-8') as f:
#     read_data = f.read()
#     Hebrew_IDs_list = re.findall(r'\s\stt[0-9]+\t', read_data)
# Hebrew_IDs_list = [line[4:-1] for line in Hebrew_IDs_list]
#
#
# Delete extra 0s at the beginning of Hebrew movie IDs. #
```

```
#
# for item in Hebrew_IDs_list:
#     if item[0] == '0':
#         Hebrew_IDs_list[Hebrew_IDs_list.index(item)] = item[1:]
# for item in Hebrew_IDs_list:
#     if item[0] == '0':
#         Hebrew_IDs_list[Hebrew_IDs_list.index(item)] = item[1:]
#
#
# Open and read files for movies with Hebrew as the primary
 ↪  language. #
#
# for dirName, subdirList, fileList in os.walk(corpus_path):
#     if len(fileList) > 0:
#         f = dirName + '/' + fileList[0]
#         folders = re.split('/', dirName)
#         if folders[len(folders)-1] in Hebrew_IDs_list:
#             find_and_count(open_and_read(f))
#
# ------------- END OF LANGUAGE-SPECIFIC BLOCK -------------
#################################################################


#################################################################
# -------------------- CALCULATIONS -------------------- #
#################################################################

# Calculate token count per corpus
for lemma in lemma_by_corpus_dict:
    for corpus in lemma_by_corpus_dict[lemma]:
        token_count_dict[corpus] = token_count_dict.get(
            corpus, 0) + lemma_by_corpus_dict[lemma][corpus]

```

```
# Calculate total frequencies per lemma
for lemma in lemma_by_corpus_dict:
    lemma_totals_dict[lemma] =
↪   sum(lemma_by_corpus_dict[lemma].values())

# Calculate total token count
for corpus in token_count_dict:
    total_tokens_int = total_tokens_int +
↪   token_count_dict.get(corpus, 0)

# Calculate DPs
for lemma in lemma_by_corpus_dict.keys():
    for corpus in lemma_by_corpus_dict[lemma].keys():
        lemma_DPs_dict[lemma] = lemma_DPs_dict[lemma] + abs(
            (token_count_dict[corpus] /
             total_tokens_int) -
            (lemma_by_corpus_dict[lemma][corpus] /
             lemma_totals_dict[lemma]))
lemma_DPs_dict = {lemma: DP/2 for (lemma, DP) in
↪   lemma_DPs_dict.items()}

# Calculate UDPs
lemma_UDPs_dict = {lemma: 1-DP for (lemma, DP) in
↪   lemma_DPs_dict.items()}


############################################################
# -------------- SORT LIST AND CREATE TABLE -------------- #
############################################################

# Sort entries by UDP
UDP_sorted_list = [(k, lemma_UDPs_dict[k]) for k in sorted(
    lemma_UDPs_dict, key=lemma_UDPs_dict.__getitem__,
```

```
150     reverse=True)]
151
152 # Create list of tuples with all values (Lemma, Frequency, Range,
    ↪ UDP)
153 for k, v in UDP_sorted_list[:list_size_int]:
154     table_list.append((k, lemma_totals_dict[k], sum(
155         1 for count in lemma_by_corpus_dict[k].values() if count >
            ↪ 0),
156         v))
157
158 ###################################################################
159 # ---------------- SORT-BY-FREQUENCY BLOCK -----------------
160 #
161 # Sort entries by raw frequency (total lemma count). To sort the
    ↪ final #
162 # list by frequency instead of UDP, comment out the above code
    ↪ within the #
163 # "SORT LIST AND CREATE TABLE" section, and also uncomment the
    ↪ relevant #
164 # lines of code in this block. #
165 #
166 #
167 # Sort entries by raw frequency #
168 #
169 # frequency_sorted_list = [(k, lemma_totals_dict[k]) for k in
    ↪ sorted(
170 #     lemma_totals_dict, key=lemma_totals_dict.__getitem__,
171 #     reverse=True)]
172 #
173 #
174 # Create list of tuples with all values (Lemma, Frequency, Range,
    ↪ UDP) #
175 #
```

```
# for k, v in frequency_sorted_list[:list_size_int]:
#     table_list.append((k, v, sum(
#         1 for count in lemma_by_corpus_dict[k].values() if count > 0),
#         lemma_UDPs_dict[k]))
#
# ------------- END OF SORT-BY-FREQUENCY BLOCK -------------
#################################################################

# Calculate list size for 80% coverage and set that as the list size. Note
# that if the initial list_size_int (set near the beginning of the script)
# provides less than the desired coverage, it will default to that instead.
#
# added_freq_int = 0
# count = 0
# for k, v in UDP_sorted_list:
#     if added_freq_int / total_tokens_int < 0.8:
#         added_freq_int = added_freq_int + lemma_totals_dict[k]
#         count = count + 1
#     else:
#         break
# list_size_int = count

# Write final tallies to CSV file
result = open('./export/HebrewWordList2.csv', 'w')
result.write('LEMMA, FREQUENCY, RANGE, UDP\n')
for i in range(list_size_int):
    result.write(str(table_list[i][0]) + ', ' +
                 str(table_list[i][1]) + ', ' +
                 str(table_list[i][2]) + ', ' +
```

```
                    str(table_list[i][3]) + '\n')
result.close()

# Print final tallies. Uncomment this code to see the results
# printed instead of writing them to a file.
#
# for i in range(list_size_int):
#     print('Lemma: ' + table_list[i][0] +
#           '\tFrequency: ' + str(table_list[i][1]) +
#           '\tRange: ' + str(table_list[i][2]) +
#           '\tUDP: ' + str(table_list[i][3]))
```

Appendix B.2: OMDb-fetch.py

```
#! /usr/bin/env python3
# -*- coding: utf-8 -*-

# import re
from sys import argv
import os
import glob
import omdb

# year = '1996'
script, year, id_start = argv

dirs = []
p = []


for name in glob.glob(
        '../OpenSubtitles2018_parsed/parsed/he/' + year + '/*/'):
    p.append(name)
# p = Path('../OpenSubtitles2018_parsed/parsed/he')
# p = list(p.glob('[198-199]*/*/*.xml'))

p = [os.path.basename(os.path.dirname(str(i))) for i in p]

for i in p:
    if i not in dirs:
        dirs.append(i)

for i in dirs:
    while len(i) < 7:
        dirs[dirs.index(i)] = '0' + i
```

```
        i = '0' + i

dirs.sort()

# for i in dirs:
#     print('tt' + i)

print('# ' + year + '\n' +
      'IMDb ID\tTitle\tYear\tLanguage(s)')


omdb.set_default('apikey', '906517b3')

for i in dirs:
    if id_start != '':
        if i > id_start:
            print('tt' + i + '\t', end="", flush=True)
            doc = omdb.imdbid('tt' + i)
            # if doc['language'] == 'Hebrew':
            print(doc['title'] + '\t' +
                  doc['year'] + '\t' +
                  doc['language'])
    else:
        print('tt' + i + '\t', end="", flush=True)
        doc = omdb.imdbid('tt' + i)
        # if doc['language'] == 'Hebrew':
        print(doc['title'] + '\t' +
              doc['year'] + '\t' +
              doc['language'])
```

Appendix B.3: single_file_extract.py

```
#! /usr/bin/env python3
# -*- coding: utf-8 -*-

import shutil
import os

source = '../OpenSubtitles2018_parsed'
destination = './OpenSubtitles2018_parsed_single'

# Copy the directory tree into a new location
shutil.copytree(source, destination,
 ↪  ignore=shutil.ignore_patterns('*.*'))

# Copy the first file in each folder into the new tree
for dirName, subdirList, fileList in os.walk(source):
    for fname in fileList:
        if fname == '.DS_Store':
            fileList.remove(fname)
    if len(fileList) > 0:
        del fileList[1:]
        src = dirName + '/' + fileList[0]
        dst = destination + dirName[27:] + '/'
        shutil.copy2(src, dst)
```

Appendix C: Movies used

References

Adelman, J. S., Brown, G. D. A., & Quesada, J. F. (2006). Contextual diversity, not word frequency, determines word-naming and lexical decision times. *Psychological Science*, *17*(9), 814–823. https://doi.org/10.1111/j.1467-9280.2006.01787.x

Albert, A., MacWhinney, B., Nir, B., & Wintner, S. (2013). The Hebrew CHILDES corpus: Transcription and morphological analysis. *Language Resources and Evaluation*, *47*(4), 973–1005. https://doi.org/10.1007/s10579-012-9214-z

Al-Surmi, M. (2012). Review: Quaglio (2009). Television dialogue: The sitcom Friends vs. Natural conversation. Philadelphia: John Benjamins. *Corpora*, *7*(1). https://doi.org/10.3366/corp.2012.0022

Amir, N., Silber-Varod, V., & Izre'el, S. (2004). Characteristics of Intonation Unit Boundaries in Spontaneous Spoken Hebrew and Acoustic Correlates. In B. Bernard & I. Marlien (Eds.), *Speech Prosody 2004, Nara, Japan, March 23-26, 2004: Proceedings* (pp. 677–680). Nara, Japan.

Anglin, J. M., Miller, G. A., & Wakefield, P. C. (1993). Vocabulary development: A morphological analysis. *Monographs of the Society for Research in Child Development*, *58*(10), i–186. https://doi.org/10.2307/1166112

Balota, D. A., & Chumbley, J. I. (1984). Are Lexical Decisions a Good Measure of Lexical Access? The Role of Word Frequency in the Neglected Decision Stage. *Journal of Experimental Psychology: Human Perception & Performance*, *10*, 340–357.

Balota, D. A., Yap, M. J., Cortese, M. J., Hutchison, K. A., Kessler, B., Loftis, B., … Treiman, R. (2007). The English Lexicon Project. *Behavior Research Methods*, *39*(3), 445–459.

Bauer, L., & Nation, P. (1993). Word families. *International Journal of Lexicography*, *6*(4), 253–279. https://doi.org/10.1093/ijl/6.4.253

Biber, D. (1988). *Variation across speech and writing.* Cambridge, UK: Cambridge University Press.

Biber, D. (1993). Representativeness in corpus design. *Literary and Linguistic Computing*, *8*(4), 243–257. https://doi.org/10.1093/llc/8.4.243

Biber, D. (1995). *Dimensions of Register Variation: A Cross-Linguistic Comparison.* Cambridge University Press.

Biber, D., Conrad, S., & Leech, G. (2002). *Longman student grammar of spoken and written English.* Harlow, Essex: Longman.

Biber, D., Johansson, S., Leech, G., Conrad, S., & Finegan, E. (1999). *Longman grammar of spoken*

and written English. Harlow: Longman.

Brezina, V., & Gablasova, D. (2015). Is there a core general vocabulary? Introducing the new general service list. *Applied Linguistics*, *36*(1), 1–22. https://doi.org/10.1093/applin/amt018

Brysbaert, M., & New, B. (2009). Moving beyond Kučera and Francis: A critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for American English. *Behavior Research Methods*, *41*(4), 977–990. https://doi.org/10.3758/BRM.41.4.977

Carroll, J. B. (1970). An alternative to Juilland's usage coefficient for lexical frequencies and a proposal for a standard frequency index. *Computer Studies in the Humanities and Verbal Behaviour*, *3*(2), 61–65.

Carroll, J. B., Davies, P., & Richman, B. (1971). *The American heritage word frequency book*. Boston: Houghton Mifflin.

Chung, T. M., & Nation, P. (2003). Technical vocabulary in specialised texts. *Reading in a Foreign Language*, *15*(2), 103.

Chung, T. M., & Nation, P. (2004). Identifying technical vocabulary. *System*, *32*(2), 251–263. https://doi.org/10.1016/j.system.2003.11.008

Collins Cobuild English grammar. (2005). Glasgow: HarperCollins.

Cowie, A. P. (2009). *The Oxford History of English Lexicography*. Oxford Univ.

Coxhead, A. (2000). A new academic word list. *TESOL Quarterly*, *34*(2), 213–238. https://doi.org/10.2307/3587951

Coxhead, A. (2016). Reflecting on Coxhead (2000), "a new academic word list". *TESOL Quarterly*, *50*(1), 181–185. https://doi.org/10.1002/tesq.287

Dang, T. N. Y., & Webb, S. (2016). Evaluating lists of high-frequency words. *ITL - International Journal of Applied Linguistics*, *167*(2), 132–158. https://doi.org/10.1075/itl.167.2.02dan

Dekel, N. (2010). *A matter of time: Tense, mood and aspect in Spontaneous Spoken Israeli Hebrew*. Utrecht: LOT.

Dice, L. R. (1945). Measures of the Amount of Ecologic Association Between Species. *Ecology*, *26*(3), 297–302. https://doi.org/10.2307/1932409

Droop, M., & Verhoeven, L. (2003). Language proficiency and reading ability in first- and second-language learners. *Reading Research Quarterly*, *38*(1), 78–103.

Ellis, N. C. (2002a). Frequency effects in language processing: A a review with implications for

theories of implicit and explicit language acquisition. *Studies in Second Language Acquisition*, *24*. https://doi.org/10.1017/S0272263102002024

Ellis, N. C. (2002b). Reflections on frequency effects in language processing: A response to commentaries. *Studies in Second Language Acquisition*, *24*, 297–339. https://doi.org/10.1017.S0272263102002140

Francis, W. N., Kučera, H., & Mackie, A. W. (1982). *Frequency analysis of English usage: Lexicon and grammar*. Boston: Houghton Mifflin.

Fries, C. C., & Traver, A. A. (1960). *English Word Lists*. Ann Arbor: George Wahr.

Gardner, D. (2007). Validating the construct of word in applied corpus-based vocabulary research: A critical survey. *Applied Linguistics*, *28*(2), 241–265. https://doi.org/10.1093/applin/amm010

Gilner, L. (2011). A primer on the general service list. *Reading in a Foreign Language*, *23*(1), 65.

Goldberg, Y. (2011, November). *Automatic Syntactic Processing of Modern Hebrew* (PhD thesis). Ben-Gurion University, Beer-Sheva, Israel.

Goldberg, Y., & Elhadad, M. (2009). Hebrew dependency parsing: Initial results. In *Proceedings of the 11th International Conference on Parsing Technologies* (pp. 129–133). Paris: Association for Computational Linguistics.

Goldberg, Y., & Elhadad, M. (2010). Easy-First Dependency Parsing of Modern Hebrew. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages* (pp. 103–107). Los Angeles, CA, USA: Association for Computational Linguistics.

Goldberg, Y., & Elhadad, M. (n.d.). Two Syntactic Parsers for Modern Hebrew and a large automatically parsed corpus.

Gretz, S., Itai, A., MacWhinney, B., Nir, B., & Wintner, S. (2015). Parsing Hebrew CHILDES transcripts. *Language Resources and Evaluation*, *49*(1), 107–145. https://doi.org/10.1007/s10579-013-9256-x

Gries, S. T. (2008). Dispersions and adjusted frequencies in corpora. *International Journal of Corpus Linguistics*, *13*(4), 403–437. https://doi.org/10.1075/ijcl.13.4.02gri

Gries, S. T. (2010). Dispersions and adjusted frequencies in corpora: Further explorations. In S. T. Gries, S. Wulff, & M. Davies (Eds.), *Corpus linguistic applications: Current studies, new directions* (pp. 197–212). Amsterdam: Rodopi.

Gries, S. T. (2017). *Quantitative corpus linguistics with R* (2nd ed.). New York: Routledge.

Guthmann, N., Krymolowski, Y., Milea, A., & Winter, Y. (2008). Automatic Annotation of Morpho-Syntactic Dependencies in a Modern Hebrew Treebank. *LOT Occasional Series*, *12*, 77–90.

Hirsh, D., & Nation, P. (1992). What vocabulary size is needed to read unsimplified texts for pleasure? *Reading in a Foreign Language*, *8*, 689–696.

Hoek, J., Evers-Vermeul, J., & Sanders, T. (2015). The role of expectedness in the implicitation and explicitation of discourse relations.

Hu, M., & Nation, P. (2000). Unknown vocabulary density and reading comprehension. *Reading in a Foreign Language*, *13*(1).

Itai, A., & Segal, E. (2003). A Corpus Based Morphological Analyzer for Unvocalized Modern Hebrew.

Izre'el, S. (2004). Transcribing Spoken Israeli Hebrew: Preliminary Notes. In D. D. Ravid & H. B.-Z. Shyldkrot (Eds.), *Perspectives on Language and Language Development: Essays in Honor of Ruth A. Berman* (pp. 61–72). Kluwer: Dodrecht. https://doi.org/10.1007/1-4020-7911-7_6

Izre'el, S., Auran, C., Bertrand, R., Chanet, C., Colas, A., Di Cristo, A., … Vion, M. (2005). Intonation Units and the Structure of Spontaneous Spoken Language: A View from Hebrew. In *Proceedings of the IDP05 International Symposium on Discourse-Prosody Interfaces* (p. 20).

Izre'el, S., Hary, B., & Rahav, G. (2001). Designing CoSIH: The Corpus of Spoken Israeli Hebrew. *International Journal of Corpus Linguistics*, *6*(2), 171–197. https://doi.org/10.1075/ijcl.6.2.01izr

Jakubíček, M., Kilgarriff, A., Kovář, V., Rychlý, P., & Suchomel, V. (2013). The TenTen Corpus Family. In *7th International Corpus Linguistics Conference CL 2013* (pp. 125–127). Lancaster.

Jang, S.-C. (1998). *Dimensions of spoken and written Taiwanese: A corpus-based register study* (Ph.D. Dissertation). University of Hawaii.

Jeon, E. H. (2011). Contribution of morphological awareness to second-language reading comprehension. *The Modern Language Journal*, *95*(2), 217–235. https://doi.org/10.1111/j.1540-4781.2011.01179.x

Juilland, A. G., Brodin, D. R., & Davidovitch, C. (1970). *Frequency dictionary of French words*. The Hague: Mouton de Gruyter.

Kilgarriff, A., & Grefenstette, G. (2003). Introduction to the special issue on the web as corpus. *Computational Linguistics*, *29*(3), 333–347.

Kobeleva, P. P. (2012). Second language listening and unfamiliar proper names: Comprehension barrier? *RELC Journal*, *43*(1), 83–98. https://doi.org/10.1177/0033688212440637

Kučera, H., & Francis, W. N. (1967). *Computational analysis of present-day American English*. Brown University Press.

Leech, G., Rayson, P., & Wilson, A. (. O. L. U. (2001). *Word Frequencies in Written and Spoken*

English: Based on the British National Corpus. Harlow: Pearson Education.

Lijffijt, J., & Gries, S. T. (2012). Correction to Stefan Th. Gries' "Dispersions and adjusted frequencies in corpora " International Journal of Corpus Linguistics 13:4 (2008), 403-437. *International Journal of Corpus Linguistics*, *17*(1), 147–149. https://doi.org/10.1075/ijcl.17.1.08lij

Lison, P., & Tiedemann, J. (2016). OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, 7.

Lyne, A. A. (1985). Dispersion. In *The vocabulary of French business correspondence* (pp. 101–124). Geneva, Paris: Slatkine-Champion.

Matsushita, T. (2012). In What Order Should Learners Learn Japanese Vocabulary? A Corpus-based Approach.

Mettouchi, A., Lacheret-Dujour, A., Silber-Varod, V., & Izre'el, S. (2007). Only Prosody? Perception of speech segmentation in Kabyle and Hebrew. In *Intefaces discours prosodie : Actes du 2ème Symposium international & Colloque Charles Bally* (pp. 207–218).

Milton, J. (2009). *Measuring second language vocabulary acquisition*. Bristol, UK ; Buffalo N.Y.: Multilingual Matters.

Mochizuki, M., & Aizawa, K. (2000). An affix acquisition order for EFL learners: An exploratory study. *System*, *28*, 291–304. https://doi.org/10.1016/S0346-251X(00)00013-0

Nagy, W., Anderson, R. C., Schommer, M., Scott, J. A., & Stallman, A. C. (1989). Morphological families in the internal lexicon. *Reading Research Quarterly*, *24*(3), 262–282. https://doi.org/10.2307/747770

Nagy, W. E., Diakidoy, I.-A. N., & Anderson, R. C. (1991). The development of knowledge of derivational suffixes. *Center for the Study of Reading Technical Report; No. 536*.

Nation, I. (1982). Beginning to learn foreign vocabulary: A review of the research. *RELC Journal*, *13*(1), 14–36. https://doi.org/10.1177/003368282013001O2

Nation, I. (2006). How large a vocabulary is needed for reading and listening? *Canadian Modern Language Review*, *63*(1), 59–82.

Nation, I. (2016). *Making and using word lists for language learning and testing*. Amsterdam: John Benjamins Publishing Company. https://doi.org/10.1075/z.208

Nation, I. S. P. (1990). *Teaching & learning vocabulary* (1 edition). Boston, Mass: Heinle ELT.

Nation, I. S. P. (2013). *Learning vocabulary in another language* (2nd ed.). Cambridge: Cambridge University Press.

Nation, I. S. P., & Webb, S. (2010). *Researching and analyzing vocabulary* (1 edition). Boston, MA: Heinle ELT.

Nation, P. (2000). Learning vocabulary in lexical sets: Dangers and guidelines. *TESOL Journal*, *9*(2), 6–10. https://doi.org/10.1002/j.1949-3533.2000.tb00239.x

Nation, P. (2004). A study of the most frequent word families in the British National Corpus. In P. Bogaards & B. Laufer (Eds.), *Language Learning & Language Teaching* (Vol. 10, pp. 3–13). Amsterdam: John Benjamins Publishing Company. https://doi.org/10.1075/lllt.10.03nat

Nation, P., & Coxhead, A. (2014). Vocabulary size research at Victoria University of Wellington, New Zealand. *Language Teaching*, *47*(03), 398–403. https://doi.org/10.1017/S0261444814000111

Nation, P., & Kyongho, H. (1995). Where would general service vocabulary stop and special purposes vocabulary begin? *System*, *23*(1), 35–41. https://doi.org/10.1016/0346-251X(94)00050-G

New, B., Brysbaert, M., Veronis, J., & Pallier, C. (2007). The use of film subtitles to estimate word frequencies. *Applied Psycholinguistics*, *28*(4), 661–677. https://doi.org/10.1017/S014271640707035X

Popescu, M., & Dinu, L. P. (2008). Rank Distance as a Stylistic Similarity. In *Coling 2008: Companion volume: Posters* (pp. 91–94). Manchester, UK: Coling 2008 Organizing Committee.

Quaglio, P. (2009). *Television dialogue: The sitcom Friends vs. Natural conversation*. John Benjamins Publishing.

Read, J. (1988). Measuring the vocabulary knowledge of second langauge learners. *RELC Journal*, *19*(2), 12–25. https://doi.org/10.1177/003368828801900202

Restrepo Ramos, F. D. (2015). Incidental vocabulary learning in second language acquisition: A literature review. *PROFILE Issues in Teachers' Professional Development*, *17*(1), 157–166. https://doi.org/10.15446/profile.v17n1.43957

Rosengren, I. (1971). The quantitative concept of language and its relation to the structure of frequency dictionaries. *Études de Linguistique Appliquée*, *1*, 103–127.

Rychlý, P. (2008). A lexicographer-friendly association score. In P. Sojka & A. Horák (Eds.) (pp. 6–9). Karlova Studánka, Czech Republic: Masaryk University.

Schmitt, N. (2010). *Researching vocabulary: A vocabulary research manual*. Houndmills, Basingstoke, Hampshire; New York: Palgrave Macmillan.

Schmitt, N., Jiang, X., & Grabe, W. (2011). The percentage of words known in a text and reading comprehension. *The Modern Language Journal*, *95*(1), 26–43. https://doi.org/10.2307/41262309?ref=no-x-route:cb78a69b6dc8bf1478b58d47243b1248

Schmitt, N., & Meara, P. (1997). Researching vocabulary through a word knowledge framework. *Studies in Second Language Acquisition*, *19*(1), 17–36.

Schmitt, N., & Schmitt, D. (2014). A reassessment of frequency and vocabulary size in L2 vocabulary teaching. *Language Teaching*, *47*(04), 484–503. https://doi.org/10.1017/S0261444812000018

Schmitt, N., & Zimmerman, C. B. (2002). Derivative Word Forms: What Do Learners Know? *TESOL Quarterly*, *36*(2), 145–171. https://doi.org/10.2307/3588328

Sima'an, K., Itai, A., Winter, Y., Altman, A., & Nativ, N. (2001). Building a tree-bank of modern Hebrew text. *Traitement Automatique Des Langues*, *42*(2), 247–380.

Sorell, C. J. (2012). Zipf's law and vocabulary. In C. Chapelle (Ed.), *The Encyclopedia of Applied Linguistics*. Hoboken, NJ, USA: John Wiley & Sons, Inc.

Sorell, C. J. (2013). *A study of issues and techniques for creating core vocabulary lists for English as an international language* (Unpublished Dissertation). Victoria University of Wellington, Wellington, New Zealand.

Sørensen, T. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. *Kongelige Danske Videnskabernes Selskab*, *5*(4), 1–34.

The history of Collins COBUILD. (n.d.). https://www.collinsdictionary.com/cobuild/.

Thorndike, E. L. (1941). *The teaching of English suffixes*. New York: Teachers College, Columbia University.

Tiedemann, J. (2012). Parallel Data, Tools and Interfaces in OPUS, 5.

Tiedemann, J. (2016). Finding Alternative Translations in a Large Corpus of Movie Subtitles, 5.

Tyler, A., & Nagy, W. (1989). The acquisition of English derivational morphology. *Journal of Memory and Language*, *28*(6), 649–667. https://doi.org/10.1016/0749-596X(89)90002-8

Tyler, A., & Nagy, W. (1990). Use of derivational morphology during reading. *Cognition*, *36*(1), 17–34. https://doi.org/10.1016/0010-0277(90)90052-L

van Zeeland, H., & Schmitt, N. (2013). Lexical coverage in L1 and L2 listening comprehension: The same or different from reading comprehension? *Applied Linguistics*, *34*(4), 457–479. https://doi.org/10.1093/applin/ams074

Wang, M., Cheng, C., & Chen, S.-W. (2006). Contribution of morphological awareness to Chinese-English biliteracy acquisition. *Journal of Educational Psychology*, *98*(3), 542–553. https://doi.org/10.1037/0022-0663.98.3.542

Ward, J., & Chuenjundaeng, J. (2009). Suffix knowledge: Acquisition and applications. *System*, *37*(3), 461–469. https://doi.org/10.1016/j.system.2009.01.004

Webb, S. A., & Chang, A. C.-S. (2012). Second language vocabulary growth. *RELC Journal*, *43*(1), 113–126.

West, M. (1953). *A general service list of English words: With semantic frequencies and a supplementary word-list for the writing of popular science and technology* (Rev. and enl. ed.). London, New York: Longmans, Green.

Whitney, P. (1998). *The Psychology of Language*. Houghton Mifflin.

Xue, G., & Nation, I. S. P. (1984). A university word list. *Language Learning and Communication*, *3*, 215–229.

Yael, M. (2014). The Haifa Corpus of Spoken Hebrew. http://weblx2.haifa.ac.il/~corpus/corpus_website/.

Zhang, H., Huang, C., & Yu, S. (2004). Distributional consistency: As a general method for defining a core lexicon. In (pp. 1119–1122). Lisbon.

Zipf, G. K. (1935). *The psycho-biology of language*. Cambridge, Mass.: M.I.T. Press.

Zipf, G. K. (1949). *Human behavior and the principle of least effort: An introduction to human ecology*. New York: Hafner.