Semantic Change in Dutch Verbs using word2vec

Master Thesis for the MA Digital Humanities

Jet Löbker

Rijksuniversiteit Groningen

Supervisor: T. Caselli

Abstract

Word embedding models such as word2vec are successful as a tool to detect and discover semantic change. However, these word2vec models are unstable due to their non-deterministic nature,In this study, I trained diachronic word embeddings using SGNS on a Dutch newspaper corpus to assess known semantic changes in verbs, automatically discover semantic changes in verbs, and evaluate the stability of the model. For this, I compiled a Dutch newspaper corpus containing 100 million words per time period, extracting the data from Delpher. Sixteen word2vec models are trained five times each, with different setting for window size and dimension size, to determine the stability of each model and the influence of these two hyperparameters. This study introduces the notion of assessing the stability in diachronic word embeddings instead of in synchronic word embeddings. The outcomes of this study support the law of conformity and present that the word2vec model is indeed unstable. The hyperparameter setting of dimension size is more influencing on stability than the setting of window size.

Contents

Introduction

Language is ever changing, and therefore it can always be researched. As every language, Dutch has changed much over time regarding its phonology, morphosyntax and semantics. For example, Old Dutch, Dutch as it was spoken before 1200, was written as:

"Hebban olla vogala nestas hagunnan hinase hic enda thu wat unbidan we nu" (*Hebben alle vogels [hun] nesten begonnen behalve ik en jij, wat beiden (= wachten) we nu?*) [1].

This passage shows sound change, as unstressed word endings have become weaker or have disappeared, e.g. 'unbidan' changed to *beiden* and 'enda' changed to *en*. It also shows semantic change, such as 'unbidan' changed to *wachten* (Steyaert, n.d.). Old Dutch also underwent morphosyntactic changes when it lost its case system. As can been seen in an example of Middle Dutch, Dutch as it was spoken between 1200 and 1500, is:

"Van dichten comt mi cleine bate. Die liede raden mi dat ict late. Ende minen sin niet en vertare." (*Het dichten geeft mij weinig baat [= voordeel]. De lieden [= mensen] raden mij aan dat ik het laat. En mijn zin [= geest] niet vermoei.*)[2]

This fragment shows how Old Dutch 'enda' changed into *en* with the intermediate stage of 'ende' in Middle Dutch. It also shows some changes in semantics, such as 'vertare' changed to *vermoei*. In contrast to the example of Old Dutch, Middle Dutch uses more words that are exactly the same as in modern Dutch, such as *dichten*, *raden* and *niet*. Other features of Middle Dutch that are displayed here are the combining of different words into one, whereby a reduced form is appended to an existing word: 'ict' corresponds to *ik 't*, and the use of a double negative 'niet en'.

How Old and Middle Dutch were spoken, depended much on where it was spoken. With the emergence of the printing press around 1450, differences in spelling, vocabulary and grammar started to disappear, but the variation in pronunciation remained. In 1637, the first written standard of the Dutch language came out with the release of the States Bible: New

[1] 'Have all birds started with their nests, except me and thee, what are we abiding now?'

[2] 'Writing poetry gives me little gain. People advise me that from it I abstain. And not wear out my brain.'

Dutch. In this form of Dutch, the combining of words, as in ’ict’, and the double negative, have become rare. Furthermore, the pronunciation of some long vowel sounds changed to diphthongs and a new reflexive pronoun ’zich’ arose.

The standardisation of Dutch continued in the sixteenth and seventeenth centuries and in 1863 the first Dutch spelling rules were adopted. In 1946, this spelling gained official status in the Netherlands and Belgium.

These examples of language change in Dutch show sound, morphosyntactic and semantic changes. In this study, semantic change will be researched, as it is, in my opinion, the most important part of language. That is, when one uses language, it is not to form phonologically correct and grammatical sentences, but to express an idea or a thought: to convey meaning.

It is assumed that words and their meanings are stored in a mental lexicon, together with their affixes and irregularities. The items in this lexicon are called lexical entries. How the lexicon is put together and how lexical entries refer to the world is not clear. There are two major approaches to this reference problem. The first, the referential theory, connects lexical entries to the real world. However, some things do not refer to the real world, such as mental states and time periods. The second approach forms an alternative by connecting lexical entries to mental representations or concepts, which in turn may be related to objects in the real world.

The semantics of a word are, among others, influenced by its part-of-speech. Every word in a sentence has a part-of-speech, which has its own semantic criteria. For example, in general a noun denotes an entity, an adverb modifies events, verbs refer to events, and determiners are function words that do not denote meaning on their own.

Much research is performed on the semantic change of especially nouns (Del Tredici, Nissim, & Zaninello, 2016; Haagsma & Nissim, 2017; Sagi, Kaufmann, & Clark, 2011). In this study, I will analyse semantic changes in verbs. The reason for this is that verbs are studied less than nouns and have clear semantic features, making them particularly interesting. Verbs can be divided in main and non-main verbs, main verbs can be divided in static and dynamic verbs, and dynamic verbs can be divided in durative and punctual verbs. Based on these semantic features, verbs denote states, activities, accomplishments, or achievements.

The study of semantic change has always relied on the analysis of large amounts of texts, collected in a corpus (Sagi et al., 2011). The intention of a corpus is to provide authentic natural language, representative for an entire language, or for a specific part of a language. The use of digital technologies is typical for corpus linguistics, as these are often developed to support or to enable to work with large amounts of data.

The now widespread availability of computational tools has opened up new methodological possibilities (Sagi et al., 2011). First, it provides objective observations, more independent of a researcher's judgement. Second, it enables researchers to observe and quantify statistical trends in large corpora without much manpower. Third, the tools can detect trends in the data based on observations on the entire corpus, that could not have been detected before.

A new computational tool that has been developed is word2vec. Word2vec is a group of computational models that produce word embeddings. The model takes a large corpus as input and produces a vector space, with each unique word in the corpus being assigned a vector. Word vectors are positioned in such a way that words with similar contexts in the corpus are located in close proximity in the vector space. When the input corpus is large enough, with enough usages and contexts of a word, word2vec can make highly accurate guesses about its meaning based on the contexts it appeared in. For example, such a guess can be tested with a word association test, e.g. 'Paris' is to 'France' what 'Amsterdam' is to '?'. Two different architectures of word2vec are the Continuous Bag-of-Words Model and the Continuous Skip-gram Model, with the skip-gram model with the extension of negative sampling (SGNS) proven to be a robust baseline (Levy, Goldberg, & Dagan, 2015; Mikolov, Chen, Corrado, & Dean, 2013; Mikolov, Sutskever, Chen, Corrado, & Dean, 2013).

This SGNS tool has also proved to be successful in diachronic language research (Haagsma & Nissim, 2017; Hamilton, Leskovec, & Jurafsky, 2016a, 2016b; Kulkarni, Al-Rfou, Perozzi, & Skiena, 2015). Word vectors of different time periods can be compared when aligned, after which semantic dissimilarity can be measured by, for example using cosine distance. A disadvantage of using a skip-gram model, is that it is based on neural networks that imply random processes and thereby show stability problems (Hellrich & Hahn, 2016a; Pierrejean & Tanguy, 2018a). When a word2vec model is trained multiple times with the exact same hyperparameters, it results in different word embeddings.

As the availability of large diachronic corpora is necessary for constructing word embeddings, most studies on semantic change have focused on English (Hamilton et al., 2016b; Hellrich & Hahn, 2016a; Pierrejean & Tanguy, 2018a). Much less research has been done on other languages, such as Swedish (Tahmasebi, 2018), Italian (Del Tredici et al., 2016), and Dutch (Haagsma & Nissim, 2017).

In their study on detection of diachronic meaning shifts in Dutch, Haagsma and Nissim (2017) used the skip-gram architecture with hierarchical softmax, with as input subcorpora with sizes ranging from 4.3 million to 35.6 million words. As the skip-gram architecture with negative sampling outperforms skip-gram with hierarchical softmax, this study adapts the skip-gram with negative sampling architecture. To use skip-gram negative sampling for learning historical embeddings on new data, Hamilton et al. (2016b) recommend to use corpora with at least 100 million words per time period. It is thus necessary for this study to compile a corpus with larger subcorpora than Haagsma and Nissim (2017) used.

In this present study, semantic changes in Dutch verbs will be studied, using subcorpora of at least 100 million words per time period. For this, the skip-gram negative sampling model will be used, with a focus on the stability of this model. This study will try to answer the following research questions:

1. Which changes does the word2vec model discover in verbs?
2. When is the word2vec model most stable regarding discovered changes?
3. How do frequent verbs change according to the word2vec model?
4. When is the word2vec model most stable regarding frequent verbs?
5. Does the word2vec model capture known historical semantic changes?

In the following chapters, an extensive literature overview will be given. Afterwards, the collection of my corpus and the methodology will be discussed. Next, the results will be presented and, finally, conclusions will be drawn from the results after which these will be discussed.

For this study, multiple python scripts are written and used. These python scripts can be found in a GitHub repository [3]. In the next chapters, I will refer to the python scripts when I used them. The python scripts are provided with comments.

[3] `https://github.com/jlobker/Thesis`

Literature review

Language change

Language is in a neverending state of change, gradually transforming over time. De Saussure already said this a hundred years ago: "Time changes all things; there is no reason why language should escape this universal law" (1916). Language change can be criticised in different ways. Language can be decaying, evolving to a more efficient state, or remaining in a similar state regarding decay or progress (Aitchison, 2013). The belief that language is deteriorating is most widespread, as seen in the thoughts on the changing use of *hun* in Dutch, when *zij* 'should' be used. This change is negatively perceived, but is, according to the director of the Taalunie, a linguistic improvement in Dutch language, as the use of *zij*, *hun*, and *hen* is based on case, while cases are no longer a functional part of Dutch (van Walsum, 2017).

Language change has a gradual implementation and a gradual spread (Aitchison, 2013). Many studies of change depend on the notion that after the language acquisition period, the grammar of an individual becomes stable (Cedergren, 1988). It is assumed that after that period the grammar of an individual does not undergo major changes, except for lexical changes. Generative grammar follows this notion in explaining language change. A distinction can be made between changes in language competence and language performance. Language competence is hidden knowledge about language stored in a grammar, or a collection of parameter settings. The product of grammar, which is used in communication, is language performance. According to the theory of generative linguistics, grammars change between two generations (Hróarsdóttir, 2004). A child uses primary linguistic data that interacts with universal grammar to build its own grammar. Changes in language performance are changes in the language that is used. That changed language then becomes input during the language acquisition of a learner, and, thus, may facilitate possible changes in language competence: changes in the grammar of a language learner. Therefore, language use changes, which leads to new input for the next generation, and so precedes changes in grammar.

Semantics and semantic change

Semantics. Semantics is the study of linguistic meaning and of how expressions convey meaning. Linguistic meaning has two aspects: sense and reference (Mihalicek & Wilson, 2011). The sense of an expression is the mental representation of its meaning, some kind of concept. The reference is the relationship of the expression to the world. Over time, sense has been described in many different ways. A very basic theory of semantics is the definitions theory, which states that dictionary-style definitions of the meaning of words should be established to give meaning to linguistic expressions (Mihalicek & Wilson, 2011; Saeed, 2015). Trying to give definitions to words shows three challenges for semantics in general. The first is circularity: to give a definition of a word, words are needed, and those words and their definitions need to be understood as well. The second challenge is defining how exact a definition has to be. Meaning is a type of knowledge, linguistic knowledge, and may differ from encyclopedic knowledge, knowledge about the way the world is. An example of Saeed (2015) is the word *whale*, as people may think it is a fish, or may think it is a mammal. The question is then whether this means that *whale* has two different meanings. The third challenge is the contribution of context to meaning, and how this contribution can be included in a definition.

Another possible form of senses is that a word's meaning is stored as a mental image. The idea is that people use mental images to conceptualise reality (Mihalicek & Wilson, 2011). However, people have different mental images, without the meaning of a word varying much. And most words do not have a mental image, like *forget* and *the*. So mental images cannot be the entire sense of a word. It appears that it is unknown what a sense precisely is, but regardless of the form of a mental representation of a word, if one knows what a word means, one knows under what conditions one can appropriately use that word (Mihalicek & Wilson, 2011).

The idea that there are conditions that determine what a word means, is also used in idea of meaning used in cognitive semantics (Bree, 1996; Traugott & Dasher, 2004). This strand of semantics states that there is a prototype of a word, an ideal representation, and that something may belong to this word to some extent. For example, when having a prototype for *bird*, like a sparrow, other concepts may also be a bird when comparing them to a sparrow. A category thus exists of a prototypical core and connected peripheral forms, that do not meet all conditions of the prototype, but meet these condition enough to be part of the category.

In the 1950s Firth and Palmer (1968) proposed that 'the time had come to try other abstractions using the larger contexts in which words are embedded' (p.18), using the context of situation as basis for the study of meaning. The meaning of language lies in the context wherein it is used, including the participants and their verbal and non-verbal behaviour, relevant objects and events, and the observed effective result in the situation (Firth & Palmer, 1968). When the context of situation is unknown, as is true for e.g. most written texts, the meaning of a word can be known by the company it keeps (Firth, 1957, A synopsis of linguistic theory, 1930-55. 168-205.) Part of the meaning of words can be indicated by characteristic collocations, such as *They are milking the cows, Cows give milk* for the meaning of *cow.* A systematic collection of collocations is valuable in lexicographical studies when an exhaustive scheme of situational contexts cannot be set up.

Verb semantics. As this research is about verb meanings, it must be mentioned that verbs have their own semantic features. The basic distinction in verbs is between main and non-main verbs (Broekhuis & Corver, 2015). Main verbs are verbs that denote certain states of affairs in which one or more participants are involved, taking *n* predicates. These verbs function as the semantic heads of their clause, meaning that one clause always contains one main verb. Non-main verbs add additional information to the expression given by the main verb and thus must always be combined with another verb. Verbs that do not function as main-verbs can be classified as perfect auxiliaries (*hebben* and *zijn*), passive auxiliary (*worden*) and modal verbs (*kunnen*, *moeten*, *mogen* and *willen*). One must keep in mind that *hebben*, *zijn* and *worden* can also be used as main verbs, and that modal verbs can occur as only verb in a sentence, but that these are exceptions to the rule that non-main verbs must be combined with main verbs.

Elements of the meaning of verbs correlate to different situation types (Broekhuis & Corver, 2015; Saeed, 2015). Verbs are either static or dynamic. Static verbs, like *zijn*, *hebben*, *weten*, and *houden van*, describe situations with no internal phases or changes. Dynamic verbs, like *rijden* and *leren*, describe situations which change over time. A further distinction can be made in dynamic verbs: they can be durative or punctual and telic or atelic. Durative verbs describe situations that last for a period of time, while punctual verbs describe events that involve virtually no time, like *niezen.* The second distinction between telic and atelic defines whether a verb describes a situation with a natural end-point (telic) or not (atelic). A telic verb is for example *bouwen*, an atelic verb *kijken.* Verbs may be inherently telic or atelic, but combining them with other elements can result in a different aspect: *Ze was aan het zingen* (atelic) vs. *Ze was een liedje aan het zingen* (telic). So, these elements classify not just the main verb, but the larger structure headed by the main verb. Vendler (1967) identified four situation types based on the semantic distinctions of verbs:

- States: static verbs (*begrijpen*, *haten*, *geloven*).
- Activities: dynamic, durative, and atelic verbs (*rennen*, *lopen*, *bibberen*).
- Accomplishments: dynamic, durative, and telic verbs (*leren*, *een kilometer lopen*, *herstellen*, *oversteken*).
- Achievements: dynamic, punctual, and telic verbs (*vinden*, *stoppen*, *beginnen*).

Semantic change. The part of language change this research is concerned with is semantic change. New words emerge, other words disappear, and existing words change in meaning. Distinctions between linguistic meanings are gradient rather than fully determined, due to their prototypical nature (Traugott & Dasher, 2004). This causes that the core of a prototype can change over time.

There is a general classification of twelve types of semantic change for organising different kinds of semantic change (Campbell, 2013). In this classification, not all categories are distinct, some overlap and intersect. The Dutch examples given originate from examples given of Dutch semantic change in Bree (1996), applied to the fitting types.

1. Widening: the range of meanings of a word increases, from more concrete to more abstract.

2. Narrowing: the range of meanings of a word decreases, from more abstract to more concrete. An example of narrowing and widening in Dutch is *pen*, that changed from 'bird feather' to 'bird feather to write with ink' (narrowing) to 'writing utensils to write with ink' (widening).

3. Metaphor: a metaphor involves understanding one thing in terms of another thing that is somehow similar in a way. In this type of semantic change, the meaning of a word is extended in such a way that there is a semantic similarity or connection between the new sense and the original one. An example in Dutch is *blad*, that changed from 'part of a tree, leaf' to 'piece of paper, sheet'.

4. Metonymy: the meaning of a word takes on additional new senses which are closely associated with the original meaning.

5. Synecdoche: this meaning change involves a part-to-whole relationship, where a part is used to refer to the whole, or the whole is used to refer to part.

6. Displacement: one word absorbs part or all of the meaning of another word with

which it is linked in a phrasal constituent. It is sometimes considered a kind of synecdoche.

7. Degeneration: the sense of a word shifts towards a more negative value in the minds of the language users. For example the Dutch *wijf*, which first had the neutral meaning 'wife, woman' and now has a negative connotation.

8. Elevation: the sense of a word shifts towards a more positive value in the minds of the language users. This happened in Dutch with *moed*, whose meaning changed from 'state of mind' to 'courage'.

9. Taboo replacement and avoidance of obscenity: in this change, the meaning remains, but the phonetic realisation gets altered, as the original form may be linked to something too obscene, or something taboo. An example in Dutch is the concept of *zijn ontlasting doen*, 'defecate', which was later expressed as *drukken*, which originally meant 'to press'.

10. Hyperbole: the meaning shifts due to exaggeration by overstatement.

11. Litotes: the meaning shifts due to exaggeration by understatement.

12. Semantic shift due to language contact.

Some of these types of semantic change in this classification can be regarded as part of one of the other changes. Therefore, others only differentiate in four types of semantic change: from abstract to concrete (narrowing); from concrete to abstract (widening); based on similarity (metaphor); and based on any kind of connection (metonymy) (Bree, 1996).

A semantic change must go through a stage of polysemy, where a word has multiple meanings (Campbell, 2013; Traugott & Dasher, 2004). A word expands its meaning, taking on additional senses and so becoming polysemous, or a word is already polysemous and looses one or more meaning(s). A view on semantic change in which polysemy is necessary, combines these situations: the words starts with an original meaning, acquires additional meanings, and then looses its original meaning. Another view on semantic change involves the idea that a word has a core meaning, and various less central meanings. In this view, in semantic change, a less central meaning becomes more central and the original core concept becomes more peripheral or lost. A third view is that meaning is a semantic map where items within a semantic domain and from other domains are related by overlapping in the polysemous choice. Semantic change follows paths of connections in this map, selecting different senses for different contexts.

The second view is similar to the view on semantic change regarded from the prototypical theory (Bree, 1996; Traugott & Dasher, 2004). In this view, a language user can use a word in a new context, due to its flexible nature. A meaning then changes when the new usage becomes accepted and widely used. Besides acquiring a new sense, meaning can also change according to this theory as notions of a word become more central or more peripheral.

A cause of semantic change is often a historical change in for example technology, society, politics, or any other part of human life. As the world is constantly changing, new or changing things or concepts need to be named, as efficiently as possible. To comply with this, it is necessary to use words for multiple concepts, so no new words need to be fabricated or borrowed. It is language efficient to express as much as possible with as few words as possible (Bree, 1996).

Some claims about semantic change are made to explain directions of change (Campbell, 2013; Traugott & Dasher, 2004). First, semantically related words often undergo parallel semantic shifts. Second, phonetic similarity can lead to shifts which leave the phonetically similar forms semantically more similar. Third, spatial words may develop temporal senses. Fourth, some common semantic shifts typically go in one direction and not the other. For example, physical-action verbs change to mental-state verbs: 'grasp' comes to mean also 'understand' and 'feel' changes from 'feel with hands' to 'feel for, think about'.

Corpus Linguistics

Corpus linguistics is an approach to studying language in which a collection of texts, assumed to be representative for natural, authentic language, is used for linguistic analysis (Jensen, 2014; Tognini-Bonelli, 2001). It is an inductive research method, as theoretical statements or conclusions are derived from observations of actual language instances found in a corpus. Corpora characteristically contain enormous amounts of texts, which can be collected and analysed using computational methods. This results in new linguistic patterns that have not been discovered earlier.

A corpus has been defined in different ways: as a collection of text, texts, whole texts, or only running texts that is in only some definitions representative and/or naturally occurring, set up for a specific purpose or linguistic analysis (Tognini-Bonelli, 2001). It is generally agreed that a corpus does not just give information on its contents, but that the results will be typical of the language of the corpus. Tognini-Bonelli (2001) defines a corpus as 'a computerised collection of authentic texts, amenable to automatic or semi-automatic processing or analysis. The texts are selected according to explicit criteria in order to capture the regularities of a language, a language variety or a sub-language' (p. 55).

Corpus linguistics covers two dimensions: the data dimension and the methodology dimension (Jensen, 2014). The data dimension involves the compilation of a corpus, and the data included in it. The methodology dimension involves the use of these data in linguistics analysis. Both dimensions adopt the digital technology, which is typical for corpus linguistics.

Some considerations in constructing a corpus are the kind of texts included, the number of texts, and the length of text samples. In order to appropriately use a corpus as a basis for generalisations concerning language, a corpus has to be representative. Representativeness 'refers to the extent to which a sample includes the full range of variability in a population' (Biber, 1993, p. 1). Typically researchers focus on sample size to achieve representativeness, but a thorough definition of the target population and decisions concerning the method of sampling are more important (Biber, 1993). The target population can be defined in (1) the boundaries of the population (what texts are included and excluded) and (2) the hierarchical organisation within the population (what text categories are included).

In choosing a sampling frame, an operational definition of the population, considerations of efficiency must be balanced against higher degrees of representativeness. When given a suitable sampling frame, a sample can be selected. All samples should be randomly selected, so that all texts in the population, or all texts within a subgroup, have an equal chance of being selected. If it is the case that the corpus must represent an entire language, a sample of all texts in the language must be included. Subgroups then need to be defined and sampled separately to obtain more representative samples. Biber (1993, p. 3) proposes a set of sampling subgroups, shown in Table 1.

The first parameter divides the corpus into three major components: writing, speech, and scripted speech, which all require different sampling considerations (Biber, 1993). So, not all subsequent parameters are relevant for each component. Within writing, a distinction that can be made is publication (2.), as the population of published texts can be operationally bounded. The parameters listed under Addressee (4.) are irrelevant for published writing, as it is always written for unenumerated, absent addressees, is non-interactive and never requires personal knowledge. Within unpublished writing, speech and scripted speech, setting (3.) can

Table 1

Subgroup parameters

1. *Primary channel.* Written/spoken/scripted speech
2. *Format.* Published/not published
3. *Setting.* Institutional/other public/private
4. *Addressee.*
   (a) Plurality. Unenumerated/plural/individual/self
   (b) Presence in place and time. Present/absent
   (c) Interactiveness. None/little/extensive
   (d) Shared knowledge. General/specialised/personal
5. *Addressor*
   (a) *Demographic variation.* Sex, age, occupation, etc.
   (b) *Acknowledgement.* Acknowledged individual/institution
6. *Factuality.* Factual-informational/intermediate or indeterminate/imaginative
7. *Purposes.* Persuade, entertain, edify, inform, instruct, explain, narrate, describe, keep records, ...
8. *Topics.*

provide sampling frames. Institutions include e.g. offices, factories, businesses, schools, churches and hospitals; other public settings include e.g. shopping and recreation centres and public media; and private settings include homes. Parameters that are important for all texts in writing, speech and scripted speech are factuality, purposes and topics.

In a sample design, the selection of texts across subgroups must be proportional to the larger population in order to be representative (Biber, 1993). A problem with a proportional corpus is that proportional samples do not represent the relative importance of a subgroup, but only its frequency. Newspapers are for example much more influential than their frequencies indicate.

The second dimension of corpus linguistics, the methodology, has two major research approaches: corpus-based and corpus-driven (Biber, 2012; Tognini-Bonelli, 2001). In corpus-based research, the corpus is used to expand, test or exemplify theories and descriptions that already exist. The starting point of research is a linguistic theory or model and the corpus elaborates on this. The goal is to discover systematic patterns of language use that define the linguistic features that are already recognised by standard linguistic theory (Biber, 2012). The corpus can also indicate minor corrections and adjustments that can be made to the theory adopted, but the evidence in a corpus is never in a position to challenge this theory (Tognini-Bonelli, 2001).

In corpus-driven research, the starting point is the observations made in a corpus, leading to hypothesis, leading to generalisation, leading to unification in a theory (Tognini-Bonelli, 2001). It is an inductive research method. The detection of linguistic phenomena should be done without prior assumptions and expectations, and only rely on the evidence provided. Without corpus-evidence, the theory does not exist. Very large, full text corpora are needed wherein new patterns can emerge. The issue of representativeness of the corpus is extremely important since a theory is based on evidence from this corpus. If a corpus turns out to be unrepresentative, errors will be made in defining a theory.

Some corpus-driven research has challenged the notion of lemma, arguing that each word form tends to have distinct meanings and uses (Biber, 2012). Even inflected variants of the same lemma are treated separately, as each form would have its own grammar and meaning. The example of 'eye' is given, as the singular form is commonly used in fixed expressions, while the plural form often refers to a body part (Biber, 2012).

Word Embeddings

Distributional semantics is a method of representing meaning. This method relies on the view that meaning comes from usage: the meaning of a word can be derived from its linguistic context (Firth & Palmer, 1968). The hypothesis of distributional models of word meanings is the idea that words that occur in similar contexts tend to have similar meanings (Clark, 2015). An example of a computational tool that uses distributional semantics are vector space models of word meaning. In vector space models, the set of contexts in which a word occurs, or the distribution of the word's context, is used to derive an appropriate meaning representation. Distributional semantic models are *count* models, they count co-occurrences among words (Ruder, 2016c).

Word embeddings that result from such vector space models, i.e. embeddings based on a word's context to represent meaning, seem successful as a tool to detect semantic change (Hamilton, Leskovec, & Jurafsky, 2016b). To implement this method for semantic change, words are embedded in vector spaces according to their co-occurrence relationships, and these embeddings are then compared across time periods (Hamilton et al., 2016b). Word embeddings can be constructed using count-based or prediction models. The following section outlines various ways in which prediction based models can be constructed for creating word embeddings.

Neural Network Models: CBOW and Skip-gram. Word vectors can be learned by neural networks, typically enabling the creation of predictions of the next word given the preceding words (Bengio, 2008; Mikolov, Yih, & Zweig, 2013). They differ from the 'original' count models, as these neural word embedding models are predicting models, that predict surrounding words. Neural networks can represent concepts locally, where one neuron is dedicated to one concept. Such a local representation is easy to understand and easy to learn, but very inefficient when data have a componential structure. A more efficient representation is a distributed representation, in which many neurons represent each concept and each neuron is part of the representation of many concepts (Hinton, 2011). A distributed word representation is a vector containing its features that characterise its meaning. It is likely that words with a similar meaning have similar vectors. These features are discovered by the learning algorithm (Bengio, 2008; Mikolov, Yih, & Zweig, 2013). The learned vectors encode many linguistic regularities and patterns (Mikolov, Sutskever, et al., 2013).

The general building blocks of a neural language model are an input layer, a hidden or projection layer or layers, and an output or softmax layer (McCormick, 2017; Ruder, 2016b). In the input layer, the words are represented as vectors that have as many components as words in the vocabulary. The word's vector has at all positions '0', except the position corresponding the current word, which has '1'. The hidden layer is a matrix with one row for every word in the vocabulary and one column for every feature. The output is a vector containing the calculated probability distribution over words in the vocabulary (McCormick, 2017).

Mikolov, Chen, et al. (2013) have introduced model architectures for learning distributed representations or embeddings of words by neural networks, both part of the word2vec tool. The word2vec is a tool for computing continuous distributed representations of words. It takes a text corpus as input and produces the word vectors as output.

The first proposed architecture is the Continuous Bag-of-Words Model (CBOW).

Instead of looking at only the preceding words for predictions, this model predicts the current word based on the preceding and following words, thus its context. Mikolov, Chen, et al. (2013) have obtained the best outcomes on correctly classifying the current word with four preceding and four future words at the input. The word order does not influence the output layer in the neural network, therefore the bag-of-words term. The second architecture introduced is the Continuous Skip-gram Model (Mikolov, Chen, et al., 2013). Instead of predicting the current word based on its context, this model predicts the context of the current word, based on the current word. The objective is to learn the vector representation of a current word that is useful for predicting the surrounding words (Mikolov, Sutskever, et al., 2013). Here, the word order does have an influence: the more distant words are, the less related they usually are, so fewer distant words are sampled.

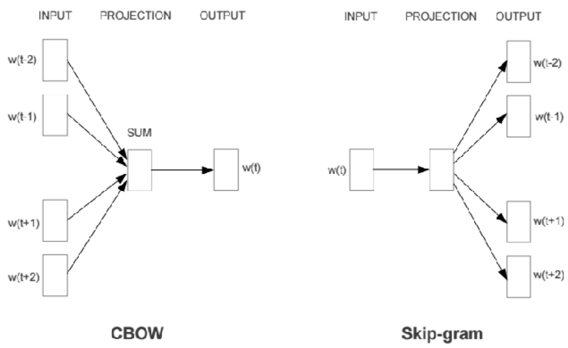


Figure 1. CBOW and Skip-gram architectures (Mikolov, Chen, et al., 2013)

The original skip-gram model can be optimised in several ways Mikolov, Sutskever, et al. (2013). The standard skip-gram formulation uses the softmax function, which calculates the probability of a word given its context Ruder (2016a). Softmax comes with high computing costs as the output embedding of every word in the vocabulary needs to be computed in order to get the probability of the target word. To perform skip-gram more efficiently, hierarchical softmax or negative sampling can be used. The hierarchical softmax uses a binary tree representation of the output layer instead of the flat softmax layer, which decreases the computing costs.

Instead of changing the softmax function, it can also be removed when using a sample-based approach, such as negative sampling. Normally when training a neural network, all vectors are slightly adjusted with each training sample. With negative sampling, each training sample only modifies a small percentage of the vectors consisting of some negative words and the current word. For smaller datasets, 5-20 words are randomly selected, and for large datasets, 2-5 words are randomly selected. These negative words are negative, because they will not fit the context (McCormick, 2017).

Another extension of the original model is the subsampling of frequent words. The most frequent words in a corpus, such as 'a' and 'the', provide little information about the current word as they appear in many contexts. Also, they have many more samples than is needed to learn a good vector. The skip-gram model thus benefits less from the co-occurrence of a word with a frequent word, than of the co-occurrence with an uncommon word. The subsampling, and thus deleting, of frequent words results in faster training and better representations of uncommon words.

When comparing different models of word embeddings on word similarity and analogy detection tasks, skip-gram with negative sampling (SGNS) proves to be a robust baseline, that does not significantly underperform in any scenario (Levy et al., 2015). SGNS performs best on discovering semantic shifts from data, based on diachronic word embeddings (Hamilton et al., 2016b). Hyperparameter tuning significantly improves performance of embeddings algorithms (Levy et al., 2015).

Hyperparameters. Modifications on the hyperparameters have a substantial impact on the performance of a word embedding algorithm. In many cases, changing the setting of a single hyperparameter has a better effect than switching to another algorithm or training on a larger corpus (Levy et al., 2015). Hyperparameters that can be tuned in an SGNS model are the following (Levy et al., 2015; Ruder, 2016c):

- Context window: the size of the context window determines how many words before and after a given word should be included as context words of the given word. Word2vec employs a scheme wherein contexts surrounding the target word are more important than contexts further away. This way, context words are weighed according to their distance from the focus word.

- Dynamic context window: with a dynamic context window, the actual window size is not fixed, but dynamic and sampled between 1 and the maximum context window.

- Subsampling frequent words: subsampling is a method of randomly deleting very frequent words. Words that are more frequent than some threshold are removed. In word2vec, the removal of these tokens is done before creating the context windows, so the context window's size is practically enlarged for many tokens. This is called dirty subsampling, opposed to clean subsampling which does not affect the context window's size.

- Deleting rare words: rare tokens are removed from the corpus, also before creating the context windows. The actual size of the context window is thus increased further.

- Shifted PMI: this is the number of negative samples. The higher the number of negative samples, the more data is being used and the better the estimation of the parameters should be.

- Context Distribution Smoothing: in order to smooth the original contexts' distribution, all context counts can be raised to the power of α. This smoothing leads to frequent words being sampled relatively less often than their frequency indicates.

- Adding context vectors: context vectors can be added to the word vectors to add first-order similarity terms to the second-order similarity function. The first-order similarity measures the tendency of one word to appear in the context of the other word, while the second-order similarity measures the extent to which the two words are replaceable as they appear in similar contexts. If context vectors are added, the similarity measure states that words are similar if they tend to appear in similar contexts, if they tend to appear in the contexts of each other, or both.

- Vector normalisation: all vectors, i.e. the rows of the matrix, are normalised to unit length. However, no normalisation can be omitted, normalisation of the columns can be used, or normalisation of both the rows and the columns can be used.

Word2vec's default values are: small context windows of 2, dynamic contexts, dirty subsampling, five negative samples, context distribution smoothing of 0.75.

Related work

Historical embeddings. Using diachronic word embeddings to study semantic change has proved to be effective (Haagsma & Nissim, 2017; Hamilton et al., 2016a, 2016b; Kulkarni et al., 2015). To construct diachronic word embeddings, embeddings are first constructed in each time period, and are then aligned over time. The semantic similarity, or distance, between two words is approximated by the cosine similarity between their vectors (Hamilton et al., 2016b).

Hamilton et al. (2016b) developed a methodology for quantifying semantic change using embeddings by comparing different count-based and prediction approaches. They apply their methodology using different corpora in four languages (English, German, French, and Chinese), so their results are not only focused on English semantics, as is in many cases. The hyperparameters for the SGNS model are tuned as followed, recommended by Levy et al. (2015) and further fine tuned by Hamilton et al. (2016b):

- a dynamic context size;
- a context window of 4;
- embeddings of size 300;
- the context distribution smoothing of 0.75;
- a negative sample of 5.

They evaluated the diachronic validity of the different approaches on the tasks of detecting known shifts and discovering shifts from data (Hamilton et al., 2016b). SGNS performed best on detecting known shifts with a large dataset (Google books with $8.5 * 10^{11}$ tokens) and on discovering shifts from data, both tested on only the English corpora. Hamilton et al. (2016b) propose two laws, following their results: (1) the law of conformity - frequent words change more slowly, and (2) the law of innovation - independent of frequency, polysemous words change more quickly. Furthermore, Hamilton et al. (2016a) show how two different computational measures can detect two different types of semantic change, changes caused by cultural shifts and changes caused by regular processes of linguistic drift. In general, nouns are more likely to undergo changes due to cultural shifts, and verbs are more likely to change due to regular processes of change (Traugott & Dasher, 2004). This difference between nouns and verbs is used by Hamilton et al. (2016a) to compare the two measures. They use the diachronic word2vec embeddings constructed in Hamilton et al. (2016b). The first measure, global distance measure, analyses the cosine distance between a word's vectors for two time periods. The second measure, local neighbourhood measure, uses second-order vectors, or context vectors (Levy et al., 2015) based on the word's nearest neighbours and not all other words in the vocabulary. The measure then analyses the local neighbourhood change, measuring the extent to which the word's similarity with its nearest neighbours has changed. Only the verbs and nouns within the top 10,000 words by frequency rank that occurred 500 times or more were examined. Their study shows that the local neighbourhood measure is more sensitive to semantic change in nouns and that the global distance measure is more sensitive to semantic change in verbs (Hamilton et al., 2016a).

Haagsma and Nissim (2017) present a study in which the semantic space of pre-selected Dutch words is compared across time, critically assessing the factors that contribute to these results, using the skip-gram architecture with hierarchical softmax. They thus use a top-down evaluation, where a set of words with known meaning shifts is selected by the authors and by editors of the Van Dale dictionary, and asses whether their method can detect these shifts. A problem with this approach is that it is unknown a priori whether these known meaning shifts actually occur in the corpus. So, the study may show that the method cannot detect the shifts, or that the known shifts are not present, but it is unknown what is the case. An alternative to this approach is the bottum-up approach, assessed in Hamilton et al. (2016b), where shifts are automatically discovered, and then evaluated to see if it were actual meaning shifts.

A second issue that is reported, is polysemy. When a word embedding for a polysemous word changes, it is hard to distinguish whether it is due to a shift in the distribution of existing senses (a more peripheral sense becomes more central or the other way around), or due to the rise of a novel sense (Haagsma & Nissim, 2017). Also, when a word already has many senses, the rise of a new sense might not cause the word embedding to shift significantly.

Quality and stability of word embeddings. An interpretation that is generally agreed upon, is that hyperparameter optimisation of word embedding models is more important than the choice of model itself (Levy et al., 2015; Sahlgren & Lenci, 2016). Sahlgren and Lenci (2016) have studied the effect of data size and frequency range of the test items on various models. They tried to answer the questions of which distributional semantics models should be opted for if there is only access to limited amounts of data, and which models should be opted for if the test items are infrequent. This second question is particularly interesting, as according to the law of conformity, frequent words change more slowly than infrequent words (Hamilton et al., 2016b). The data sizes studied are that of a corpus with 1 million, 10 million, 100 million, and 1 billion words. The chosen embedding size is 200 dimensions and the chosen context window size is 2 words. Their results show that SGNS does not produce competitive results for the data sets smaller than 1 billion words, but that the model's results do steadily improve as the data set becomes bigger (Sahlgren & Lenci, 2016). Also, none of the other distributional semantics models performs well on the small data sets, with these parameters chosen. When testing for the impact of frequency, they split the vocabulary of the 1 billion corpus into a high range part, with frequencies over 16,830, medium range part, with frequencies between 16,795 and 729, and low range part, with frequencies under 728. SGNS here scores best on the high frequency range, and does not underperform much on the other frequency ranges.

Not only the quality of the word embeddings can change due to parameter choices, also the stability of word embeddings is not a given fact. The term stability refers to the consistency of the neighbourhood of words around a given target word: a stable word has the same nearest neighbours across several models (Chugh, Whigham, & Dick, 2018; Pierrejean & Tanguy, 2018a). When retraining a model with the exact same hyperparameters, words neighbourhoods may change because of the inherent randomness of the algorithm (Pierrejean & Tanguy, 2018a). The word2vec model is thus non-deterministic: the algorithm can exhibit different behaviours on different runs, even with the same input and the same hyperparameters. An element that makes this model non-deterministic are the weights in the hidden layer, which are randomly initialised, causing the run of the model to proceed from a different point in the vector space (Chugh et al., 2018).

Pierrejean and Tanguy (2018a) try to predict the stability of a word based on a set of selected features, the features examined are among others: part-of-speech, degree of polysemy, frequency, entropy, and cosine similarity of the word nearest neighbour. Pierrejean and Tanguy (2018a) measure the stability for a word between two embeddings models by measuring the nearest neighbours overlap for words common to the two models, using 25 neighbours. They use the same parameters in all models, namely the SGNS algorithm with negative sampling rate of 5, context window size of 5, dimensions of 100, number of iterations of 5 and minimal count of 100. The used corpora have a size of 100 million words.

The study of Pierrejean and Tanguy (2018a) shows that the variation score of a word can vary between zero, where all 25 nearest neighbours are identical, and 0.68, where only a third of the nearest neighbours are found in both models. The feature that has the most impact on the stability of a word is the cosine similarity of the word nearest neighbour. A second feature with high impact is part-of-speech, proper nouns have a higher variation than other categories, followed by nouns. Furthermore, words with very high or very low frequency are less stable than words in the mid-frequency range; the more polysemic a word is, the more it is likely to be unstable; and the higher the entropy score of a word, indicating a high variability in contexts, the less stable it is.

Chugh et al. (2018) also studied the stability of word vectors by comparing the neighbourhood of words in the word embedding model. They consider the impact of frequency of words on stability, and the impact of word embedding dimension size. In this study, the stability is measured by calculating the Jaccard similarity index between the pairs of sets of neighbours generated during different runs of the model. It returns values in the range of zero, no matching, to one, both sets are identical, so values closer to one indicate greater consistency in the neighbourhood of a given word. The experiment was conducted on three corpora, using only the 10,000 most common tokens. The tokens were split according to their frequency into the ten most frequent words, the ten least frequent words, and the words in between. They then created a word2vec model with 100 iterations and window size of 2, for each dimension size ranging from 1 to 377. Finally, the ten nearest neighbours of the words are saved using the different word embeddings and the Jaccard index is calculated.

The results of this study are that high frequency words show greater stability and that the optimal embedding dimension size for maximum stability is corpus dependent (Chugh et al., 2018). The vocabulary size and frequency distribution of words will determine what dimension size is needed.

Pierrejean and Tanguy have not only measured the stability of words using two embeddings models with the same hyperparameters, but also the stability between different word embedding models trained with different parameters (Pierrejean & Tanguy, 2018b). They compare nineteen models, by using the parameters of their default model and training the other models by changing the value of one parameter at a time. They also identify features for words remaining stable independently of the parameter values used. The influence of parameter setting is measured by the variation in neighbours between two models. The parameters that are studied are architecture (skip-gram versus CBOW), corpus (BNC versus ACL), window size (5 versus 1 to 10), vectors dimensions (100 versus 50 to 600), context type (window based versus dependency based).

This second study of Pierrejean and Tanguy shows that changing parameters creates not significant difference in models performance, but that there is much variation between the different models compared to the default model Pierrejean and Tanguy (2018b). Even for models varying the least, the variation has an average of at least 0.3, meaning that by changing one parameter, among the 25 nearest neighbours about 1 out of 3 is different from one model to the other. Changing the architecture and changing the corpus results in the highest variation between these models and the default model. The models with less drastic differences with the default model, when the vector size was changed from 100 to 200 or the window size from 5 to 6, show the lowest variation.

Hellrich and Hahn (2016a, 2016b) have investigated both accuracy, or quality, and reliability, or stability, of the skip-gram word embedding algorithms. They have compared continuous training of models, where the model for each time period is initialised with the embeddings of the previous time period, and independent training, where each time period is trained on its own and the models are then mapped (Hellrich & Hahn, 2016a). Next, they focused on the difference between hierarchical softmax and negative sampling (Hellrich & Hahn, 2016a, 2016b). They evaluated accuracy by using a test set that contains groups of four words with a similarity relation, such as king is to queen as man is to woman. Reliability is evaluated by training three models with the exact same hyperparameters and then comparing the closest neighbours using the Jaccard similarity index.

The first study of Hellrich and Hahn shows that the independent training is superior to the continuous training, and that negative sampling is more accurate and more reliable than hierarchical softmax in this independent training (Hellrich & Hahn, 2016a). To further explore this approach, the influence of the number of training epochs, word frequency and word ambiguity are measured in the first and the second study to see if negative sampling still performs better than hierarchical softmax (Hellrich & Hahn, 2016a, 2016b). With the negative sampling, reliability increases with each epoch, but maximal accuracy is achieved after only 2 epochs, with additional epochs leading to a small decrease. After the fifth epoch in Hellrich and Hahn (2016a), negative sampling outperforms hierarchical softmax in reliability. In Hellrich and Hahn (2016b) both corpora in English and German are attested. Their research shows that the number of necessary epochs for negative sampling to outperform hierarchical softmax is linked to both language and corpus size. For word frequency, negative sampling is overall more reliable than softmax, in particular for words with low or medium frequency. Also when looking at the number of senses of a word, or the ambiguity, negative sampling performs better.

The quality and stability of word embeddings over time is tested on a Swedish corpus, by Tahmasebi (2018). The goal is to investigate whether a word embedding method can be effectively used on a corpus of limited volume and quality. She focuses on eleven words that either represent stable concepts, new concepts, or have the potential to reveal cultural changes. Her corpus contains over 5 million tokens per year, for the years 1749-1925. The word2vec model is run for each year separately. To study whether a word embedding is stable, she took the ten nearest words of the investigated words per year and calculated the Jaccard similarity between each pair of adjacent years. She concluded that the higher the frequency of a word, the higher the stability for the vectors and the lower the frequency, the lower the stability is.

Data Collection

When constructing a representative corpus, it is important to consider what kind of texts, what number of texts and which length of texts to include (Biber, 1993). These considerations should be balanced against considerations of efficiency. Following Hamilton et al. (2016b), I put together a corpus that contains 100 million words per time period of ten years. To be able to do this efficiently, I chose to collect written texts from newspapers, which is the most extensive collection of online available Dutch language. I thus favour efficiency and feasibility of compiling a corpus over a corpus being perfectly representative for an entire language. However, Biber (1993) states that newspapers are more influential, and thus representative of language, on language than their frequencies as part of language as a whole indicate. I thus believe that I do, to some degree, compile a representative corpus for Dutch language.

When considering the subgroups of Biber (1993) presented in the literature review, my corpus will consist of written, published texts. The parameter of factuality of newspaper texts is most often factual-informational, but this might differ in for example opinion editorials, and the purposes of newspaper texts are mostly to inform, explain or describe.

The corpus for this research will be extracted from the newspaper section of Delpher. Delpher is a digital archive, developed by the National Library of the Netherlands, containing digitised texts of over 1.3 million Dutch newspapers, 320,000 books, and 4.4 million magazine pages from the 15th to the 21st century. I chose to only include newspapers, as this collection is by far the most extensive collection in Delpher, and contains enough data to compile a 100 million word corpus per time period. The newspapers in Delpher contain news articles, family notes such as births, deaths and marriages, illustrations, and advertisements. Examples of these text types, retrieved from an edition of Limburgsch dagblad on 30-04-1948, are the following:

- Article[4]:

```
Geen krentenbrood voor zelfverzorgers
DEN HAAG, 29 April. - Het C.D.K. heeft t.a.v. het verstrekken krentenbrood geen
regelingvoor de zelfverzorgers ontworpen, omdat zulks in de praktijk onmogelijk
is. Er zal uitsluitend krentenbrood mogen worden afgeleverd op de daarvoor
```

[4] http://resolver.kb.nl/resolve?urn=ddd:010416002:mpeg21:a0004:ocr

`aan te wijzen broodbonnen. Misschien zal de hoeveelheid toe te wijzen krenten en rozijnen de bakkers toestaan daarvan iets te verstrekken aan de zelfverzorgers, zonder hun andere klanten tekort te doen.`

- Family note[5]:

`Geven met grote vreugde kennis van de geboorte van ons dochtertje en zusje Marijke, dat bij het H. Doopsel de namen ontving van MARIA LUCIA W. J. Ploem C. Ploem-Straeten Marlies Kerkrade, 28 April 1948 Marktstraat 3 Tijd.: St. Joseph Ziekenhuis, Paviljoen, kamer 2. Met grote vreugde geven wij kennis van de geboorte van ons zoontje en broertje Huppie, dat bij het H. Doopsel de namen ontving van HUBERTUS JOHANNES ANTONITJS F. W. Berkers M. ...`

- Illustration[6]:

`4 Geslachten te Hop hoven. J. Jongkind 78 jaar, Adr. Jongkind 52 jaar, Joh. Jongkind 24 jaar en A. Jongkind 2 en een half jaar.`

- Advertisement [7]:

`LIMBURGSCHE BOEK- EN KUNSTHANDEL Saroleastraat 35 HEERLEN Tel. 3616 (K. 4440) .loris van den Bergh „TEMIDDEN DER KAMPIOENEN" met een voorwoord van Karel J. J. Lotsy Een boek voor ieder liefhebber van de wielrensport.' Naam Adres'. Plaats wenst te ontvangen: .... ex J. v.d. Bergh: „Temidden der Kampioenen" a f 6.90 Handtekening`

As the examples show, news articles present text closest to natural language. Therefore, the corpus that I compile, will only include news articles.

I chose to collect data from the time periods 1945-1954 and 1985-1994. The reason for this is that the National Library has different selection criteria for different time periods, namely 1618-1800, 1800-1814, 1814-1869, 1869-1940, 1940-1945, and 1945-1995. Collecting all newspapers from the same time period ensures that all newspapers are selected in a similar way. I selected the newest period, because the orthography of Dutch changed the least during that period, compared to the earlier periods (Geerts, Van Den Broeck, & Verdoodt, 1977; Steyaert, n.d.). The subcorpora contain texts from ten years, as multiple years were needed to

[5] http://resolver.kb.nl/resolve?urn=ddd:010416002:mpeg21:a0132:ocr

[6] http://resolver.kb.nl/resolve?urn=ddd:010416002:mpeg21:a0066:ocr

[7] http://resolver.kb.nl/resolve?urn=ddd:010416002:mpeg21:a0013:ocr

attain 100 million words. The two different time periods of 1945-1954 and 1985-1994 are selected, because this is as much time difference in between the subcorpora as possible, so that there is the most chance of demonstrable language change.

Delpher

The National Library selects the newspapers that are included in Delpher according to different political, social, economic and cultural criteria. The following criteria were set up for newspapers dated from 1945 to 1995 (Koninklijke Bibliotheek, n.d.-a):

1. the newspaper as part of the breakthrough of popular culture and consumption society;
2. the newspaper as most important forum of the political debate;
3. the newspaper as leading role in the following domains:
   (a) church and religion
   (b) ethics, morality, education
   (c) arts and critics
   (d) social relations
   (e) migration and multicultural relations
4. the newspaper as expression of new collective identities
5. and other criteria associated with the development of the journalistic profession and the newspaper sector.

After the selection procedure, the digitising of a specific newspaper depends on the availability of copies, the quality of the microfilms, and copyright (Koninklijke Bibliotheek, n.d.-b). [8]. Concerning the 1945-1954 and 1985-1994 time periods, respectively 107,410 and 35,998 newspapers are available.

Delpher uses Optical Character Recognition (OCR) to transfer scans of newspapers to texts readable for computers. The results of OCR processing might be far from perfect, especially for older texts. As a result, it is hard to judge whether a pattern in the data

[8] A list of available newspapers can be found on `https://www.kb.nl/sites/default/files/docs/Beschikbare_kranten_alfabetisch.pdf`

represents a interesting finding or whether it is a result of a systematic error in OCR (Traub, Van Ossenbruggen, & Hardman, 2015). It is possible that the subcorpus including texts from 1945-1954 contains more OCR errors than the subcorpus including texts from 1985-1994. Delpher mentions that the older the material, the harder it is to process the material with OCR: old spelling, complex page layout, tricky fonts, discolouration of the paper and fading of the ink need to be taken into account (Delpher, n.d.). However, Delpher does not present any details on how correctly the texts are digitised with OCR, and thus how reliable the texts on Delpher are for research.

Scraping Delpher

To compile news articles from Delpher, I wrote a python script to collect data per year for a decade per time period (delpherscraper.py). Using this script, I collected ten million words per year to get 100 million words per time period, so the words are equally divided in each time period. Delpher has 33,363 newspapers in its collection for the year 1945, but for 1985, it has only 3,783 newspapers. By gathering data per year instead of per decade, the time periods are comparable, and I avoid the chance of accidentally comparing only one year from the first time period with ten years from the second time period, as the amount of data is skewed.

To collect the data stored in Delpher, I used an application programming interface (API) for searching the collection. This search API allows searching issues or articles in the historical newspaper collection, based on the Search and Retrieval via URL protocol. The base URL for searching is `http://jsru.kb.nl/sru/sru`, and the queries can be specified by URL parameters to find the correct issues or articles. The result of a SRU query is an xml document with records that contain metadata of the newspaper issues or articles.

In my python script, the base URL is adjusted to search for 1000 records, which is the maximum, in the newspaper collection within the first of January and the 31st of December of a specific year. After scraping the first 1000 records, the next 1000 records are scraped, and this process continues until connection with the database is lost, or until I disconnect it. This URL results in an XML document that shows metadata of the 1000 records, and thus of 1000 newspapers.

Part of this metadata is a metadatakey, which containes the word *services*, which links to an XML document containing the metadata of one of the 1000 newspapers. Every record in this second XML document contains information about a news article, advertisement, family note, or illustration. Part of this information is what type of text it is. As I only want to collect news articles for my corpus, I only look for the texts that have *artikel* as subject. The data of these particular records contain a URL to the OCR processed title, if applicable, and text of that record. An example of such data, again from Limburgsch Dagblad on 30-04-1948, is:


```
<text>
<title>VERSLAGBOEK VAN DE 16e SOCIALE STUDIEWEEK.</title>
<p>
Btj het Limburgsch Dagblad te Heerlen verscheen het verslagboek van de 16e
Limburgse Sociale Studieweek te Rolduc. Jan Maenen constateert in. het
voorwoord, dat de lezing van alle Inleidingen, die op deze eerste na-oorlogse
studieweek gehoudenwerden - slechts de les van prof. dr. Kors kon niet
opgenomen worden -, een „deugddoende herinnering en verfrissing" zal zijn
voor degenen, die deze studieweek bezochten. Alle anderen zij kennisneming
van deze gedegen woorden over het onderwerp „Katholicisme en Neutraliteit" van
harte aanbevolen.
</p>
<p>
Het werk bevat nu de Inleidingen van aalmoezenier K. Roncken, dr. Cornelissen,
dr. Jos. de Boer, dr. N. Devolder O.P.M., dr. A. Olierook, Jos. Maenen, pater
J. Colsen en pater Jac. Jacobs. Het is verlucht met fraaie, actuele foto’s.
</p>
</text>
```


The titles and texts from this third XML document for all news articles in all newspapers are then saved in a text file. The python script results in one text document containing at least 10 million words, processed with OCR, for one year. In total the script is run twenty times, each time to collect 10 million words for one year.

Pre-processing data

The twenty text documents that resulted from the scraping of Delpher, one per year for two decades, contain between 10,320,895 and 47,186,105 tokens. To get an evenly distributed subcorpus of 100 million words per decade, all one-year texts should be about 10 million words. The text also needs to be preprocessed, to correctly train a word2vec model.

To achieve this, a second python script is used (kortetekst_TC.py). In this script, the sentences are first tokenised using nltk sent_tokenize, which divides a string into a list of substrings, in this case dividing a text into a list of sentences. This is a necessary step, as the word2vec module in gensim needs an input of one sentence per line to train a model (Rehurek, n.d.). After this, the sentences are tokenised again, this time with the spacy module, to obtain individual tokens. The text is split in sentences using nltk and in words using spacy, as nltk is faster in sentence tokenising and spacy is faster in word tokenising (Schroll, n.d.).

The resulting list of tokens is preprocessed. First, all words need to be lower cased. This needs to be done, because python makes a distinction between lower cased and upper cased words. For example, the words *kamer* and *Kamer* are not recognised as the same token, but as unique tokens, so when constructing a model, *kamer* and *Kamer* would possibly have other word vectors. Second, only the tokens that consist of alphabetic characters are kept. This way, the punctuation is not be saved, as punctuation does not have a use when constructing word embeddings. Also words containing non-alphabetic characters are not be saved. Because of the OCR processing, alphabetic characters might be read as non-alphabetic characters, resulting in odd words, such as *!aat*, what should be *gaat* [9].

The tokens that are now separated and preprocessed, can be counted. The sentences are tokenised until the list of tokens is not shorter than 10 million anymore, so when the list is 10 million tokens long, the script stops and saves the output in a new file. This new file now contains around ten million words and has one sentence per line. The files of ten consecutive years, so the files of the years 1945-1954 and 1985-1994, are pasted together in one file to create two files, or subcorpora, containing about 100 million words of one decade each. The subcorpus of the years 1945-1954 contains 99,999,882 words, and the subcorpus of the years

[9] Found in: `https://www.delpher.nl/nl/kranten/view?query=&facets%5Bperiode%5D%5B%5D=2%7C20e_eeuw%7C1940-1949%7C1945%7C&page=1&coll=ddd&resultscoll=dddtitel&identifier=MMNIOD05%3A000099646%3Ampeg21&resultsidentifier=MMNIOD05%3A000099646%3Ampeg21`

1985-1994 contains 99,999,917 words.

To extract the verbs from the text, part-of-speech tagging is applied. I used Alpino, a dependency parser for Dutch, to obtain the POS-tags [10]. As it took multiple hours to parse thousand sentences, and the subcorpora contain 6.7 and 6.9 million sentences, I only parsed 5000 random sentences of each subcorpus. I expect that the relative frequencies of verbs in these random obtained sentences match the relative frequencies of verbs in the entire subcorpus.

Corpus description

The diachronic corpus for this study on semantic change in Dutch verbs between 1945-1954 and 1985-1994 using word2vec consists of 200 million words, divided in two subcorpora of 100 million words each. Each subcorpus holds 10 million words per year, for a decade. The corpus contains newspaper articles including their titles if present, scraped from Delpher. This corpus is unannotated: it is simple plain text. The metadata of the news articles included is not saved, so information as in which newspaper on which date the articles were published is lost.

Type/token ratio. The first subcorpus, containing news articles from 1945 to 1954, contains 99,999,882 tokens and 81,272 types, thus distinct words. The second subcorpus, containing news articles from 1985 to 1994, contains 99,999,917 tokens and 69,291 types. The type/token ratio (TTR), a measure of lexical diversity, of the first subcorpus is 0.000812, and of the second 0.000693, so the second subcorpus is lexically less diverse than the first. The higher the TTR, the more lexical diverse the text is and thus the less repetitive the vocabulary usage is (Richards, 1987). The type/token ratio normally is lower the longer the text is, therefore the type/token ratio indicates relatively little lexical variation in this subcorpus.

From each subcorpus, 5000 sentences were POS-tagged to identify verbs to construct samples. The 1945-1954 sample has 10,580 verb tokens and 2,437 verb types, which gives a TTR is 0.230. The 1985-1994 sample has 9,422 verb tokens and 2,309 verb types, which gives a TTR is 0.245. These type/token ratios cannot be compared with the ratios of all part-of-speech in the previous paragraph. This comparison is uneven, as a TTR of the entire subcorpus is compared with a TTR of a sample.

---

[10] I used the command: Alpino end_hook=frames -parse

When looking at the TTR of all part-of-speech in the samples, the 1945-1954 sample has 75,343 tokens and 15,691 types, and thus a TTR of 0.208. The 1985-1994 sample has 66,975 tokens and 15,009 types, and a TTR of 0.224. So, in the sample the second sample has a higher lexical diversity than the first sample for all part-of-speech and for the verbs. As the difference in TTR of all part-of-speech in the samples, between the first and second subcorpus, is not the same as the difference in the subcorpora, I cannot state anything about how the TTR of the verbs might be, and might relate to each other, in the entire subcorpora.

Most frequent words. To get an idea of the content of the subcorpora, I collected the most frequent words. Table 2 shows the most frequent words with their relative frequencies of the subcorpora. Stop words were removed from the text before collecting these frequent words, as stop words are very commonly used words that do not contribute any real meaning that distinguishes one text from another. The stop words used are the ones provided by the NLTK corpus, complemented with all individual letters, and *we*, *wij* and *jij*, as *ik*, *je*, *hij* and *zij* were already included. The list of stop words can be found in appendix A.

Words that are very frequent in both subcorpora are *uur* and *jaar*. When looking at the texts, *uur* seems to be used a lot when describing the duration of an event. This might be typical of newspaper articles, since a purpose of news articles is to describe events, and this description naturally contains a temporal aspect. An example of this usage of *uur* is:

```
overschreed om uur te voet de bij eede aan de z grens van zeeuws vlaanderen
```

This can be roughly translated to: 'crossed the southern border of Zeeuws-Vlaanderen at o'clock by foot'. Because only alphabetic characters are kept in the text, the specific time, probably stated by a number, is lost, as are the dashes.

Likewise, *jaar* is used to describe the duration of an event, but it is also used to describe when events happened, again common for news articles:

```
het hoofd van de katholieke kerk de paus komt in mei van dit jaar naar ons toe
```

('The head of the Catholic church, the Pope, comes to us in May of this year').

The remaining frequent words give more information about the time periods, such as *mei* in 1945-1954, but presumably also about the newspapers included in the corpora. As the news articles are collected consecutively in order of the Delpher database, it is possible that the database starts with many newspapers originating from Limburg in the first time period, and from Amsterdam in the second time period. This may also explain the occurrence of *joodse* and *joden* in the second time period, as Amsterdam has always been the centre of Jewish life in the Netherlands. These frequently occurring words thus illustrate the possible skewness in the texts, due to the manner in which the data is collected.

Besides the most frequent words, I also collected the most frequent verbs in the sample of 5000 sentences that are tagged with part-of-speech labels. Table 3 shows the most frequent verbs per sample of each subcorpus with their frequencies. The samples of the subcorpora are not of the same size, the first sample has 75,343 tokens and the second sample has 66,975 tokens, so the frequencies cannot be compared between the different time periods, but only within a time period. The meanings of these verbs can be studied to consider the law of conformity, according to which frequent words change more slowly (Hamilton et al., 2016b).

In general, the meaning of lemmas is studied, even though some corpus-driven research challenges the notion of lemma, as each word form might have distinct meanings and uses (Biber, 2012). Table 3 lists these word forms. It is interesting to see if it holds that word forms change independently from each other, which can be done by comparing the difference between changes in these forms and changes in lemmas. Therefore, Table 4 shows the most frequent verb lemmas. Alpino gives the first person singular of every verb when performing the part-of-speech tagging. These first person singulars are changed to their corresponding lemma in Table 4.

In the literature review, I made a distinction between main and non-main verbs. Verbs that are, apart from some exceptions, non-main verbs, are modal verbs (Broekhuis & Corver, 2015). As main verbs express the core meaning of a clause, their semantic change is most interesting to track. The Dutch modal verbs are: *kunnen, willen, mogen, hoeven, moeten* and *zullen.* Table 5 shows the ten most frequent non-modal verbs. Table 4 and table 5 both show very little differences in frequently occurring verb lemmas, which indicates that modal verbs are not the most frequently occurring verbs.

The twelve verbs that are presented in Table 5 are *zijn, worden, hebben, komen, gaan, maken, geven, doen, houden, zeggen, zien* and *krijgen.* Verbs belong to one of the following situation types: states, activities, accomplishments and achievements (Vendler, 1967). *Zijn*, *hebben* and *houden* are states, *doen*, *zeggen* and *zien* are activities, *komen*, *gaan* and *maken* are accomplishments, and *worden*, *geven* and *krijgen* are achievements. However, these types do also depend on the context the verb is used in. For example, 'I see something' does not have a natural end-point, but 'I see this movie' does have a natural end-point. Activities can thus also be accomplishments, dependent on the context.

Table 2

The most frequent words per subcorpus

Time period	Most frequent words
1945-1954	*uur*(211.430), *jaar*(197.941), *heerlen*(195.540), *mei*(157220), *onze*(139946), *zullen*(138025), *heer*(136543), *grote*(127627), *alle*(126703), *welke*(126529)
1985-1994	*joodse*(241.942), *uur*(200.380), *jaar*(196.898), *amsterdam*(190.048), *joden*(138.500), *wel*(136.341), *alle*(130.245), *onze*(119.497), *nederland*(119.229), *twee*(102.310)

Table 3

The ten most frequent verb forms per sample of each subcorpus

Time period	Most frequent verb forms
1945-1954	*is*(638), *zijn*(356), *worden*(322), *zal*(277), *heeft*(254), *was*(249), *werd*(245), *hebben*(189), *wordt*(161), *zullen*(110)
1985-1994	*is*(733), *zijn*(336), *heeft*(279), *worden*(252), *was*(224), *hebben*(163), *wordt*(143), *werd*(123), *zal*(116), *kan*(110)

Table 4

The ten most frequent verb lemmas per sample of each subcorpus

Time period	Most frequent verb lemmas
1945-1954	*zijn*(1411), *worden*(810), *hebben*(594), *zullen*(504), *kunnen*(243), *moeten*(179), *komen*(148), *gaan*(112), *maken*(95), *geven*(94)
1985-1994	*zijn*(1419), *hebben*(612), *worden*(568), *zullen*(301), *kunnen*(239), *moeten*(184), *komen*(125), *willen*(117), *gaan*(108), *maken*(104)

Table 5

The ten most frequent non-modal verb lemmas per sample of each subcorpus

Time period	Most frequent non-model verb lemmas
1945-1954	*zijn*(1411), *worden*(810), *hebben*(594), *komen*(148), *gaan*(112), *maken*(95), *geven*(94), *doen*(93), *zeggen*(91), *houden*(83)
1985-1994	*zijn*(1419), *hebben*(612), *worden*(568), *komen*(125), *gaan*(108), *maken*(104), *zeggen*(97), *zien*(97), *doen*(79), *krijgen*(75)

Data quality

The quality of the data depends on the quality of the OCR translations from pictures to text. Figure 2 shows the scan and the OCR translation of the newspaper article 'DE BETUWE GEZUIVERD' from 'Vrije pers: voor Leiden en omstreken', 04-04-1945 [11].

To roughly examine the quality of the OCR translations in the data that are collected, I compiled a subsample of 200 tokens for both time periods [12]. With checking these words, there are three different possibilities:

1. The word is correct;
2. The word is a word in the text, but read incorrectly;
3. The word is not a word in the text.

To check which possibility applies, I read the list of 200 tokens. Words that are normal looking words are counted as (1.) correct, for example *boekhandel.* I look up words that seem to be incorrect in the entire text to find the context words. I then search this resulting string in Delpher and look for the correct newspaper, there is in almost all cases only one newspaper that matches the search and falls in the right time period. Delpher shows the original image of the newspaper with the corresponding OCR text, and here I can check if the word in the OCR text matches the word in the image. An example is the word *hi.* I assume it is unlikely that this word is used in a newspaper around the time period 1945-1954, so I find the context, which is *lijk hi ste eisch.* Delpher finds the corresponding newspaper (Limburgsch dagblad,

[11] `https://resolver.kb.nl/resolve?urn=ddd:010449568:mpeg21:a0008`

[12] I used the command: cat [subcorpus] | tr ' ' '\n' | gshuf -n200

10-05-1946) and the image shows that the text should be *-lijk herstel eerste eisch.* In other, more common, cases, only one letter is translated incorrectly. This word is then counted as actual text but not as correctly translated.

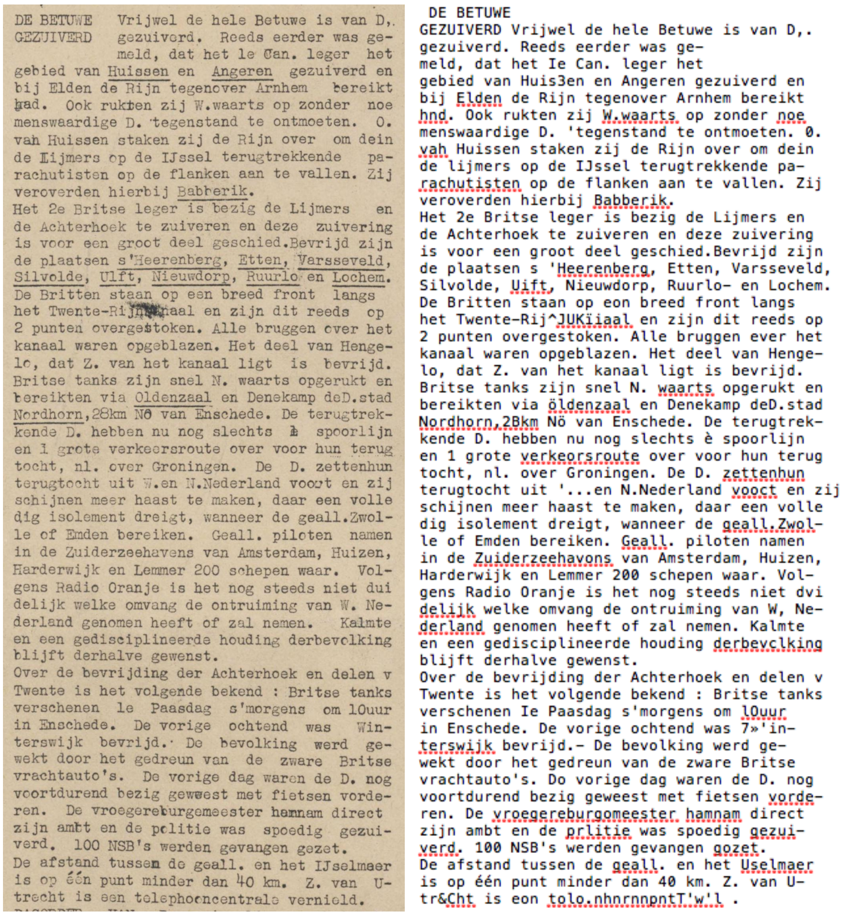
The third option, when something is read as text, when in fact it is not, occurs very little, but happens for example when something that is identified as text, is in fact e.g. part of the colouring of an image. I checked this the same way as I did with words covered by (2.). This OCR mistake occurred when I did not only collect newspaper articles, but the entire newspaper content, so also text that goes with illustrations for examples. Words that are mistaken in this third way do not count as actual words, and thus also not as correct translations.

Table 6 shows that the OCR tool identifies all sampled words except one correctly as words. The one mistake here was the letter *a*, which was in reality just a printed line at the side of a paper. In the subsample of the first time period 184 tokens, 92%, are correctly translated. The quality appears to be a little better in the sample of the second time period, where 189 tokens, 94,5%, are correctly translated. In some incorrect translated words, just one letter or two letters are inaccurate, as in *tesen* instead of *tegen* and *fxansen* for *fransen.* However, other words are unrecognisable, such as *oiivucl* instead of *singel.* These results suggest that the quality of OCR is quite high, but not perfect.

Table 6

The amount of actual and correctly translated words in a subsample of 200 tokens

Time period	Actual text	Correct translation
1945-1954	200	184 (92%)
1985-1994	199	189 (94.5%)



DE BETUWE GEZUIVERD Vrijwel de hele Betuwe is van D,. gezuiverd. Reeds eerder was gemeld, dat het le Can. leger het gebied van Huissen en Angeren gezuiverd en bij Elden de Rijn tegenover Arnhem bereikt had. Ook rukten zij W.waarts op zonder noe menswaardige D. 'tegenstand te ontmoeten. O. van Huissen staken zij de Rijn over om dein de Lijmers op de IJssel terugtrekkende parachutisten op de flanken aan te vallen. Zij veroverden hierbij Babberik.
Het 2e Britse leger is bezig de Lijmers en de Achterhoek te zuiveren en deze zuivering is voor een groot deel geschied.Bevrijd zijn de plaatsen s'Heerenberg, Etten, Varsseveld, Silvolde, Ulft, Nieuwdorp, Ruurlo en Lochem. De Britten staan op een breed front langs het Twente-Rijnkanaal en zijn dit reeds op 2 punten overgestoken. Alle bruggen over het kanaal waren opgeblazen. Het deel van Hengelo, dat Z. van het kanaal ligt is bevrijd. Britse tanks zijn snel N. waarts opgerukt en bereikten via Oldenzaal en Denekamp deD.stad Nordhorn,28km NO van Enschede. De terugtrekkende D. hebben nu nog slechts 1 spoorlijn en 1 grote verkeersroute over voor hun terug tocht, nl. over Groningen. De D. zettenhun terugtocht uit W.en N.Nederland voort en zij schijnen meer haast te maken, daar een volle dig isolement dreigt, wanneer de geall.Zwolle of Emden bereiken. Geall. piloten namen in de Zuiderzeehavens van Amsterdam, Huizen, Harderwijk en Lemmer 200 schepen waar. Volgens Radio Oranje is het nog steeds niet dui delijk welke omvang de ontruiming van W. Nederland genomen heeft of zal nemen. Kalmte en een gedisciplineerde houding derbevolking blijft derhalve gewenst.
Over de bevrijding der Achterhoek en delen v Twente is het volgende bekend : Britse tanks verschenen le Paasdag s'morgens om 10uur in Enschede. De vorige ochtend was Winterswijk bevrijd.· De bevolking werd gewekt door het gedreun van de zware Britse vrachtauto's. De vorige dag waren de D. nog voortdurend bezig geweest met fietsen vorderen. De vroegereburgemeester hernam direct zijn ambt en de politie was spoedig gezuiverd. 100 NSB's werden gevangen gezet.
De afstand tussen de geall. en het IJselmeer is op één punt minder dan 40 km. Z. van Utrecht is een telephooncentrale vernield.

DE BETUWE
GEZUIVERD Vrijwel de hele Betuwe is van D,.
gezuiverd. Reeds eerder was ge-
meld, dat het Ie Can. leger het
gebied van Huis3en en Angeren gezuiverd en
bij Elden de Rijn tegenover Arnhem bereikt
hnd. Ook rukten zij W.waarts op zonder noe
menswaardige D. 'tegenstand te ontmoeten. 0.
vah Huissen staken zij de Rijn over om dein
de lijmers op de IJssel terugtrekkende pa-
rachutisten op de flanken aan te vallen. Zij
veroverden hierbij Babberik.
Het 2e Britse leger is bezig de Lijmers en
de Achterhoek te zuiveren en deze zuivering
is voor een groot deel geschied.Bevrijd zijn
de plaatsen s 'Heerenberg, Etten, Varsseveld,
Silvolde, Uift, Nieuwdorp, Ruurlo- en Lochem.
De Britten staan op eon breed front langs
het Twente-Rij^JUKiiaal en zijn dit reeds op
2 punten overgestoken. Alle bruggen ever het
kanaal waren opgeblazen. Het deel van Henge-
lo, dat Z. van het kanaal ligt is bevrijd.
Britse tanks zijn snel N. waarts opgerukt en
bereikten via öldenzaal en Denekamp deD.stad
Nordhorn,2Bkm Nö van Enschede. De terugtrek-
kende D. hebben nu nog slechts è spoorlijn
en 1 grote verkeorsroute over voor hun terug
tocht, nl. over Groningen. De D. zettenhun
terugtocht uit '...en N.Nederland vooct en zij
schijnen meer haast te maken, daar een volle
dig isolement dreigt, wanneer de geall.Zwol-
le of Emden bereiken. Geall. piloten namen
in de Zuiderzeehavons van Amsterdam, Huizen,
Harderwijk en Lemmer 200 schepen waar. Vol-
gens Radio Oranje is het nog steeds niet dvi
delijk welke omvang de ontruiming van W, Ne-
derland genomen heeft of zal nemen. Kalmte
en een gedisciplineerde houding derbevclking
blijft derhalve gewenst.
Over de bevrijding der Achterhoek en delen v
Twente is het volgende bekend : Britse tanks
verschenen Ie Paasdag s'morgens om lOuur
in Enschede. De vorige ochtend was 7»'in-
terswijk bevrijd.- De bevolking werd ge-
wekt door het gedreun van de zware Britse
vrachtauto's. Do vorige dag waren de D. nog
voortdurend bezig geweest met fietsen vorde-
ren. De vroegereburgomeester hamnam direct
zijn ambt en de prlitie was spoedig gezui-
verd. 100 NSB's werden gevangen gozet.
De afstand tussen de geall. en het Uselmaer
is op één punt minder dan 40 km. Z. van U-
tr&Cht is eon tolo.nhnrnpntT'w'l .

(a) Image of news article. (b) OCR translation of news article.

Figure 2. Newspaper article 'DE BETUWE GEZUIVERD' from 'Vrije pers: voor Leiden en omstreken', 04-04-1945.

Methodology

To study the semantic change in Dutch verbs using word2vec, I followed the studies of (Hamilton et al., 2016a, 2016b) by constructing word embeddings in each time period using skip-gram negative sample, and then aligning them over time. The semantic similarity between two words is approximated by the cosine similarity between their vectors: the global distance measure. I used a bottum-up approach, where shifts can be automatically discovered. In addition to this, I also tracked the change of some pre-selected words, selected by Haagsma and Nissim (2017) or selected due to their frequencies.

The stability of the word embeddings is evaluated by training the word2vec model in five runs per time period with the exact same hyperparameters, following the chosen hyperparameters and part of the methodology of Pierrejean and Tanguy (2018a). I also trained models with other hyperparameters, changing the window size and the dimension size, to see which hyperparameters result in the most stable model, following Pierrejean and Tanguy (2018b). This resulted in different models, with in each model the value of one parameter changed, and with five runs per time period per model. For each model five alignments are made, aligning the embeddings of the different time periods per run.

I analysed the semantic changes found in the five alignments of each model. When the same semantic changes are detected, I considered the model as stable. This differs from how the term stability is used in most studies, where stability refers to the consistency of the neighbourhood of words around a given word (Chugh et al., 2018). I chose to assess stability by analysing the consistency of the changes in verbs, as this is a study on semantic change.

Embedding algorithm

To train word embeddings, I used the gensim Python library with the word2vec skip-gram method (Hamilton et al., 2016b; Rehurek, n.d.). I wrote the python script 1embeddings.py for this.

Word embeddings for the time period 1945-1954 are trained, and word embeddings for the time period 1985-1994. The selected hyperparameters of the default model are the hyperparameters from Pierrejean and Tanguy (2018a), because their study also used corpora of approximately 100 million words. The architecture of this default skip-gram training algorithm is used with negative sampling of 5 words, dimensionality of the vectors set to 100, window size set to 5, minimal count set to 100, number of iterations set to 5 and down-sampling rate set to 10e-3. The models from both time periods are trained five times, to construct five separate sets of word embeddings, which are then all saved.

The models with the changed parameters deviated from the default model on the value of one parameter at a time, changing the window size and the vector dimension size. I trained one model per possible parameter value stated in 7. This generates 15 models, next to the default model.

Table 7

Parameter values used to train word2vec embeddings

Parameters	Default value	Tested values
Window size	5	1 to 10
Dimension size	100	50, 200, 300, 400, 500, 600

The first parts of the word vector of the verb *word* in 1945-1954 show that the word vector is not equal in all runs from the default model:

- Run 1: [-0.23750828, -0.6127142, 0.089558005, -0.3657844, 0.38186902, ...]
- Run 2: [-0.22626337, -0.7366935, 0.27820003, -0.5131348, 0.6917693, ...]
- Run 3: [-0.3360143, -0.79178536, 0.14967172, -0.45704353, 0.7575506, ...]
- Run 4: [-0.188741, -0.8587096, 0.2839303, -0.45279008, 0.7219134, ...]
- Run 5: [-0.4030846, -1.0673695, 0.40850067, -0.48580894, 0.63174, ...]

Alignment

The trained models are aligned to the same coordinate axes between time periods to make them comparable (Hamilton et al., 2016b). The embeddings are not naturally aligned due to the randomness in nature of SGNS. I used Ryan Heuser's port of Hamilton's code to align the gensim models between time periods [13] (2alignments.py). The five runs of each model are aligned, which results in five alignments.

To make the process of comparing the embeddings as straightforward as possible, I created a Pandas dataframe for every alignment containing the words that are included in

[13] `https://gist.github.com/quadrismegistus/09a93e219a6ffc4f216fb85235535faf`

both time periods, the corresponding embeddings of the 1945-1954 time period and the corresponding embeddings of the 1985-1994 time period. The Python code I used to create this is 3dataframe.py. An example of part of one of the dataframes is shown in Table 8. I have attempted to add information about the part-of-speech to the dataframe, so verbs can be extracted conveniently. For this, I compiled a list of all verbs tagged in the 5000 sentences from 1945-1954 and in the 5000 sentences from 1985-1994. If a word in the dataframe was on this list of verbs, it got a 'verb' tag in the fourth column of the dataframe. However, when checking if the verbs in the 'word' column had been tagged or not, it appeared that many verbs were missing on the list. For example, *aanbeland*, *dineerde*, *doorbreken*, *lanceerden*, *zweren* and *zwijgt* were not tagged as verbs. Because so many verbs are not tagged, I decided to not include this column in the dataframe, but to distinguish between verbs and other words by hand in a later stage.

Table 8

The dataframe containing the word vectors for 'word' and 'aanbieden' in the alignment of run 1 of the default model

Word	Vector 1945-1954	Vector 1985-1994
word	[-0.16597115, 0.20226079, ...]	[-0.089459, -0.067642875, ...]
aanbieden	[-1.1128038, 0.012788644, ...]	[0.032357097, 0.024656644, ...]

In the following sections, I use the term alignment to refer to the combination of the aligned embeddings of the time period 1945-1954 and of the time period 1984-1995. As there are five runs per model, there are five alignments per model.

Verbs

To study semantic changes in verbs, I compiled a list of verbs which are known to have changed in meaning throughout history, although not specifically between 1945 and 1994. These verbs are *krijgen*, *bieden*, *breken*, *bewegen* and *doen* (Haagsma & Nissim, 2017; Landsbergen, 2009; Verhagen, 1998).

Krijgen has changed from a concrete, agentive verb in early middle Dutch, spoken in the 14th century, to an abstract, non-agentive verb in present-day Dutch (Landsbergen, 2009). A change that occurred between the 19th and the 20th century is the appearance of the use of sentences with an abstract object, as 'answer' and 'consent'. Another change is the rise of the number of sentences with inanimate subjects in the 20th century. This semantic change in the 20th century might arise in this research. *Krijgen* belongs to the semantic category of achievements.

Bieden, *breken* and *bewegen* are obtained from a pre-selected list of words which gained a new, figurative meaning between 2000 and 2016, composed by editors of the Van Dale dictionary, complemented with suggestions of possible candidates of semantic change between 1994 and 2016 by Haagsma and Nissim (2017). Since the studied time period by Haagsma and Nissim (2017) immediately follows the period that is studied in this research, I assume that changes in these verbs might emerge in this research as well. *Bieden* and *breken* belong to the semantic category of achievements and *bewegen* belongs to activities.

Finally, *doen* was used as causitive verb with animate causers in the 18th century which changed to inanimate causers in the 20th century (Verhagen, 1998). This is related to the level of authority of the animate causer: in the 18th century, *doen* was mainly used with an animate causer when that causer was an authority. The change indicates that in the 20th century, authority no longer is an important aspect of interpersonal relationships, in the way that we do not write about them. *Doen* is a verb that belongs to the semantic category of activities.

I also compiled a list of the twelve most frequent verb lemmas[14] in the subsets of 5000 sentences per time period with their frequencies in the 1945-1954 and 1985-1994 texts, see Table 5. To obtain the relative frequencies of these lemmas in the 200 million words corpus, I have compiled a frequency list of all words per time period, and added the frequencies of all word forms belonging to the same lemma. For example, for the frequency of the lemma *zijn*, I have added the frequencies of *ben, bent, is, zijn, was, waren, geweest* of both time periods. According to the law of conformity of Hamilton et al. (2016b), frequent words change more slowly. I wanted to analyse if this can be identified in these most frequent lemmas, as the frequency varies between 4.5 million (of *zijn*) and almost two hundred thousand (of *krijgen*). Table 9 shows these most frequent verb lemmas with their absolute frequencies from the 1945-1954 and 1985-1994 corpus.

Many of these frequent verbs are probably so frequent, because they are also part of

[14] *zijn, worden, hebben, komen, gaan, maken, geven, doen, houden, zeggen, zien* and *krijgen*

compound verbs. This is the case with *hebben*[15], *komen*[16], *gaan*[17], *maken*[18], *geven*[19], *doen*[20], *houden*[21], *zeggen*[22], *zien*[23], *krijgen*[24]. Therefore it is possible that forms, other than the infinitive and past participle, are part of a lemma that is not one of the twelve most frequent lemmas. However, as word2vec builds word embeddings for a word form, independent of the corresponding lemma, the embedding for e.g. *maak* is build from all its uses and contexts. The compound verbs of which the lemmas are part in the corpus are in the footnotes. I found these by searching for the *[infinitive] and the *[past particle] in the word list, consisting of the words that occur 100 times of more in both the 1945-1954 corpus and the 1985-1994 corpus.

To study the changes of the fifteen lemmas mentioned [25], I looked at the changes of all associated word forms. For example, for *bewegen*, I analysed the changes in *bewegen*, *beweeg*, *beweegt*, *bewoog* and *bewogen*.

In addition to looking at the change of the five verbs with known changes and of the twelve most frequent lemmas, I also tested if the algorithm can discover new shifts, by examining the top ten verbs that changed the most between 1945-1954 and 1985-1994.

[15] *liefhebben, plaatshebben*

[16] *aankomen, bijeenkomen, binnenkomen, heenkomen, meekomen, nakomen, neerkomen, omkomen, opkomen, overeenkomen, overkomen, tegenkomen, terugkomen, toekomen, uitkomen, voorkomen, voortkomen, vrijkomen*

[17] *aangaan, doorgaan, heengaan, ingaan, meegaan, nagaan, omgaan, ondergaan, opgaan, overgaan, samengaan, tegengaan, teruggaan, uitgaan, voorafgaan, voorbijgaan, voorgaan, voortgaan*

[18] *afmaken, bekendmaken, kennismaken, losmaken, meemaken, opmaken, overmaken, schoonmaken, uitmaken, vrijmaken*

[19] *aangeven, afgeven, doorgeven, ingeven, opgeven, overgeven, prijsgeven, teruggeven, toegeven, uitgeven, vrijgeven*

[20] *aandoen, afdoen, meedoen, omdoen, opdoen*

[21] *aanhouden, achterhouden, afhouden, bijhouden, ophouden, vasthouden, volhouden, weerhouden*

[22] *opzeggen, toezeggen*

[23] *aanzien, afzien, inzien, opzien, uitzien*

[24] *binnenkrijgen*

[25] *bewegen*, *bieden*, *breken*, *doen*, *gaan*, *geven*, *hebben*, *houden*, *komen*, *krijgen*, *maken*, *worden*, *zeggen*, *zien*, *zijn*

Table 9

The twelve most frequent verb lemmas in the subsets of both time periods with their absolute frequencies in the entire corpus

Lemma	Absolute frequency	Lemma	Absolute frequency
zijn	4,457,479	*geven*	290,207
worden	1,846,057	*doen*	276,048
hebben	978,986	*houden*	242,543
komen	454,627	*zeggen*	239,724
gaan	369,489	*zien*	227,779
maken	322,476	*krijgen*	183,929

Measuring semantic change

To measure semantic change, I used the global measure proposed by Hamilton et al. (2016a) as this measure is more sensitive to semantic changes in verbs. For this measure, I used a word's vector for both time periods and measured the cosine distance between them:

$$d^G(w_i^{(t)}, w_i^{(t+1)}) = cos - dist(w_i^{(t)}, w_i^{(t+1)}) \tag{1}$$

which translates to:

$$d^G(w_i^{(t)}, w_i^{(t+1)}) = 1 - \frac{w_i^{(t)} \cdot w_i^{(t+1)}}{\| w_i^{(t)} \| \| w_i^{(t+1)} \|} \tag{2}$$

In equation 1 and 2, $w_i^{(t)}$ and $w_i^{(t+1)}$ are the word vectors in respectively the first time period of 1945-1954 and the second time period of 1985-1994, $w_i^{(t)} \cdot w_i^{(t+1)}$ is the dot product between $w_i^{(t)}$ and $w_i^{(t+1)}$ and $\| w_i^{(t)} \|$ and $\| w_i^{(t+1)} \|$ are the norm of the vectors $w_i^{(t)}$ and $w_i^{(t+1)}$.

The equation results in a number between 0 and 1, with 0 meaning that the cosine of the angle between the two vectors of a word is zero, and that there is thus no difference between the two vectors and that the word has not changed between the two time periods. When the cosine distance is 1, the vectors have no match, and thus the word has completely changed between the time periods.

The cosine distance is measured for all five alignments of each model. I used a Python script to compile Python dictionaries of both time periods containing 'word: embedding' pairs.

Using these dictionaries, I measured the cosine distance with equation 2, which I then saved to a CSV file (4cosinedistance.py).

Stability of embeddings

As word2vec models are known to be unstable (Chugh et al., 2018), I compiled 16 different models, all with the value of one parameter different. As Table 7 explains, next to the default model, there is one model with the window size set to 1, one model with the window size set to 2, one model with the dimension size set to 50, and so on. I changed not more than one parameter per model, so either the window size or the dimension size was changed.

Evaluating changes

To test whether the models discover semantic shifts, I examined the top ten verbs that changed most from around the 1950s to around the 1990s per model. Following Hamilton et al. (2016b), I limited my analysis to words with relative frequencies above 10^{-5} in both decades, which translates to a thousand words in my 100 million words corpus. To allow for this limitation, I combined the CSV files containing the words and their cosine distances with the frequencies of the words in both time periods. I then sorted the cosine distances from high to low and searched for the ten most changing verbs.

When going through the words for verbs, I only classified a word as verb when that word can only be a verb, and not another part-of-speech. When in doubt, I checked the meaning of the word in the Cornetto demo (Vossen et al., 2013). Examples of word that can be verbs, but do not have to be, and thus I have not included in my results, are: *ren* ('I run' versus 'a chicken run'), *stem* ('I vote' versus 'a voice'), *luid* ('I ring' versus 'loud'), *vlucht* ('I flee' versus 'a flight'), *leer* ('I learn' versus 'leather'), *beleefd* ('experienced' versus 'polite'), *verzoeken* ('to request' versus 'multiple requests'), *dansen* ('to dance' versus 'multiple dances'), and *besluiten* ('to decide' versus 'decisions').

Searching the ten most changing verbs per run per model, resulted in a list of fifty changed verbs per model. The more duplicate verbs are in this list, the more stable this model is, as the same verb then occurs in multiple runs of the same model. To assess the stability of the model, I also took into account how many verbs occur in three or more, so more then half, runs of a model, and the variation in the degree of change indicated as cosine distance across the five runs.

Next, I studied the cosine distances of the verb forms corresponding to the fifteen verb lemmas mentioned in Verbs per model. For the twelve most frequent verbs [26], I assessed the stability of changes in these verbs, to see if the most stable model for the automatically discovered changes, is also the most stable model for frequently occurring verbs, and if the degree of polysemy as an effect on this. For the five verbs [27] that are selected because of their known changes, I checked the degree of the cosine distances of these lemmas, using the sixteen different models.

[26] *zijn, worden, hebben, komen, gaan, maken, geven, doen, houden, zeggen, zien* and *krijgen*

[27] *krijgen, bieden, breken, bewegen* and *doen*

Results

In this section I will first discuss the results of the automatically detected changes by the word2vec model, followed by the results of the changes in the fifteen aforementioned verbs. I will discuss the results for each model, with one parameter value changed at a time, followed by outcomes regarding the detected changes and stability of the sixteen models.

Automatically detected changes

This subsection presents the results of the automatically detected changes by the sixteen models and the stability of these changes. To achieve this, the ten most changing verb forms are extracted per run per model and discussed per model. For each model, I discuss the amount of different verbs changed in these top tens in each model, the amount of verbs that change in three or more runs of the model and the corresponding cosine distances with their mean and standard deviation. To assess the stability of the models, I compare the aforementioned results. Appendix B includes the results of the ten most changing verbs per model, detected by the model, and the amount of runs they changed in, per model.

Models. ***Default model.*** In the default model[28], 27 verbs appear in the top ten most changing verbs according to the five runs of the model. Table 10 presents the most changing verbs per run. The cosine distances of the 27 changed verbs range from 0.884 (*ontstaan* in run 2) to 0.983 (*luister* in run 2), with a mean cosine distance of 0.921 and a standard deviation of 0.022. Seven of the 27 verbs undergo changes in three or more runs (N), reported with their mean cosine distance (M) and standard deviation (SD): *gegroeid* (N=4, M=0.926, SD=0.019), *groeide* (N=4, M=0.918, SD=0.029), *onderscheiden* (N=4, M=0.933, SD=0.011), *volgen* (N=4, M=0.938, SD=0.020), *behandelt* (N=3, M=0.926, SD=0.011), *keert* (N=3, M=0.929, SD=0.024) and *luister* (N=3, M=0.942, SD=0.039).

Model with window size 1. According to the WIN1 model[29], 25 verbs change the most between 1945-1954 and 1985-1994, Table 11 shows which verbs change in which run. The cosine distances of the 25 changed verbs range from 0.858 (*speelt* in run 4) to 0.984 (*zond* in

[28] Hyperparameters: skip gram, dimension size=100, window size=5, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[29] Hyperparameters: skip gram, dimension size=100, window size=1, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

run 5), with a mean distance of 0.888 and a standard deviation of 0.029. Eight of these verbs change in three or more runs of this model: *wilden* (N=5, M=0.906, SD=0.017), *zond* (N=5, M=0.932, SD=0.045), *hef* (N=4, M=0.916, SD=0.039), *opgewonden* (N=4, M=0.876, SD=0.027), *antwoordde* (N=3, M=0.869, SD=0.006), *gewend* (N=3, M=0.889, SD=0.002), *luidt* (N=3, M=0.869, SD=0.003) and *ophouden* (N=3, M=0.880, SD=0.011).

Table 10

The ten most changing verbs per run for the default model

Run	Ten most changing verbs
1	*bekeken, volgen, ontmoeten, groeide, bleken, kijkt, geboden, luister, schrijft, draaide*
2	*bekeken, keert, groeide, luister, onderscheiden, inzien, gegroeid, eindigde, bood, ontstaan*
3	*gegroeid, onderscheiden, behandelt, vroegen, inzien, keert, volgen, ontbreekt, lachen, groeide*
4	*luister, volgen, onderscheiden, behandelt, gegroeid, brak, keert, nagelaten, lijdt, groeide*
5	*volgen, onderscheiden, ontbreekt, gegroeid, behandelt, wint, lachen, raden, nagelaten, leen*

Table 11

The ten most changing verbs per run for the WIN1 model

Run	Ten most changing verbs
1	*omvat, wilden, gewend, kiest, verkeert, ophouden, zond, ontsnapt, vertoont, hef*
2	*hef, opgewonden, zond, ophouden, gewend, wilden, verkeert, omvat, luidt, schreven*
3	*hef, wilden, zond, gewend, antwoordde, wint, opgewonden, luidt, ophouden, dalen*
4	*zond, wilden, hef, int, verschijnt, schaken, antwoordde, opgewonden, levert, speelt*
5	*zond, binnengekomen, wilden, opgeven, luidt, beoogt, rijdt, verschijnt, antwoordde, ogpewonden*

Model with window size 2. Table 12 shows the ten most changing verbs per run for the WIN2 model[30]. In this model 23 different verbs change between the time periods, of which eight change in three or more runs of the model. The cosine distances of the 23 changing verbs range from 0.858 (*breken* in run 1) to 0.973 (*omvat* in run 2). The mean cosine distance is 0.904 and the standard deviation is 0.025. These verbs are: *besteld* (N=5, M=0.923, SD=0.014), *gerechtigd* (N=5, M=0.937, SD=0.029), *kiest* (N=5, M=0.904, SD=0.018), *omvat* (N=4, M=0.923, SD=0.037), *uitgeven* (N=4, M=0.893, SD=0.019), *bedacht* (N=3, M=0.896, SD=0.011), *koken* (N=3, M=0.898, SD=0.025) and *zond* (N=3, M=0.901, SD=0.006).

Model with window size 3. In the model with window size set to three, WIN3[31], 32 different verbs change in the top tens of the five runs, which are presented in Table 13. The most changing verbs have a mean cosine distance of 0.916 with a standard deviation of 0.028 and vary between 0.878 (*probeert* in run 3) and 0.984 (*verkeert* in run 2). Three words change in three or more runs: *bewaren* (N=3, M=0.890, SD=0.009), *luister* (N=3, M=0.938, SD=0.019) and *volgt* (N=3, M=0.926, SD=0.040).

Model with window size 4. In the WIN4 model[32], 38 verbs change the most in the five runs, presented in Table 14. The range of the cosine distances of the verbs is from 0.866 (*gewezen* in run 4) to 0.966 (*uitbrak* in run 5), with a mean cosine distance of 0.907 and a standard deviation of 0.026. As in the previous model, three verbs change in three or more runs: *dreigde* (N=3, M=0.933, SD=0.025), *rusten* (N=3, M=0.905, SD=0.021) and *uitbrak* (N=3, M=0.952, SD=0.019).

[30] Hyperparameters: skip gram, dimension size=100, window size=2, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[31] Hyperparameters: skip gram, dimension size=100, window size=3, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[32] Hyperparameters: skip gram, dimension size=100, window size=4, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

Table 12

The ten most changing verbs per run for the WIN2 model

Run	Ten most changing verbs
1	*binnengekomen, kiest, omvat, besteld, gerechtigd, gepasseerd, wint, schieten, gegooid, breken*
2	*omvat, gerechtigd, besteld, koken, kiest, zond, verwijderd, wint, belooft, uitgeven*
3	*gerechtigd, uitgeven, omvat, besteld, kiest, woont, koken, luister, bedacht, draagt*
4	*gerechtigd, besteld, bedacht, woont, draagt, zond, uitgeven, kiest, leidt, koken*
5	*besteld, uitkomen, gerechtigd, toonde, zond, kiest, bedacht, aangegeven, uitgeven, omvat*

Table 13

The ten most changing verbs per run for the WIN3 model

Run	Ten most changing verbs
1	*gelopen, volgt, studeerde, dien, tellen, trouwen, geschoten, ophouden, droeg, ingenomen, opgeheven*
2	*verkeert, ingegaan, volgt, geschoten, gelopen, trouwen, legt, eist, gold, bewaren*
3	*speelden, legt, ingegaan, luister, dien, ondergaan, opgeheven, meldde, bewaren, volgt*
4	*vroegen, eet, opgegeven, opnemen, luister, gebroken, brengt, eist, springen, probeert*
5	*opnemen, luister, vroegen, probeert, ontwikkelen, groeit, eet, bewaren, schieten, vervolgt*

Table 14

The ten most changing verbs per run for the WIN4 model

Run	Ten most changing verbs
1	*droeg, stelden, bleven, aanvaarden, ophouden, volgt, bevrijd, opnemen, rusten, opgewonden*
2	*verdragen, meldde, geleid, richt, zwemmen, zingen, verdwijnen, draagt, opgeleverd, ontbrak*
3	*bedacht, uitbrak, dreigde, behandelt, leen, bood, geslaagd, bestaan, stoppen, ontwikkeld*
4	*uitbrak, groeide, dreigde, bedacht, behandelt, aanvaarden, rusten, zendt, neemt, gewezen*
5	*uitbrak, dreigde, neemt, rusten, zendt, groeide, verkeert, leed, verschenen, schiet*

Model with window size 6. In the model with window size set to 6, WIN6[33], 28 verbs change the most in five runs, presented in Table 15. These verbs have cosine distances between 0.900 (*aangegeven* in run 2) and 0.986 (*uitbrak* in run 4), with a mean of 0.931 and a standard deviation of 0.021. Of these 28 verbs, 7 change in three or more runs: *ontworpen* (N=5, M=0.930, SD=0.024), *opgeheven* (N=4, M=0.932, SD=0.024), *treffen* (N=4, M=0.938, SD=0.013), *behandeld* (N=3, M=0.965, SD=0.027), *inzien* (N=3, M=0.922, SD=0.012), *leert* (N=3, M0.920, SD=0.016) and *uitbrak* (N=3, M=0.959, SD=0.025).

Model with window size 7. 26 words change in the WIN7 model[34], shown in Table 16. These words have a cosine distance ranging from 0.905 (*ontmoet* in run 3) to 0.974 (*luister* in run 1), with a mean cosine distance of 0.936 and a standard deviation of 0.019. As in the previous model, 7 verbs change in three or more runs: *behandeld* (N=5, M=0.952, SD=0.005), *onderscheiden* (N=4, M=0.945, SD=0.011), *volgen* (N=4, M=0.944, SD=0.027), *begrepen* (N=3, M=0.925, SD=0.018), *luister* (N=3, M=0.937, SD=0.032), *toont* (N=3, M=0.945, SD=0.020) and *brak* (N=3, M=0.937, SD=0.015).

[33] Hyperparameters: skip gram, dimension size=100, window size=6, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[34] Hyperparameters: skip gram, dimension size=100, window size=7, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

Table 15

The ten most changing verbs per run for the WIN6 model

Run	Ten most changing verbs
1	*voert, treffen, ontworpen, bood, bedacht, heten, opgegeven, inzien, schaken, neerleggen*
2	*behandeld, inzien, treffen, afgeleverd, opgeheven, leert, volgt, bood, ontworpen, aangegeven*
3	*behandeld, opgeheven, bleken, ontworpen, treffen, leert, uitbrak, bestrijden, inzien, gelopen*
4	*uitbrak, behandeld, heette, bezorgd, verzorgen, afgeleverd, treffen, ontworpen, opgeheven, zendt*
5	*uitbrak, opgegeven, verdwijnen, betrekken, voorzien, ontworpen, bezorgd, luister, heette, leert*

Table 16

The ten most changing verbs per run for the WIN7 model

Run	Ten most changing verbs
1	*luister, lachen, gevormd, behandeld, begroepen, draait, draaien, genoten, wint, brak*
2	*volgen, overgegaan, onderscheiden, behandeld, brak, draait, toont, begrepen, haalde, eist*
3	*volgen, behandeld, onderscheiden, toont, verbieden, overgegaan, genoten, wint, begrepen, ontmoet*
4	*toont, behandeld, getrokken, brak, onderscheiden, onderzocht, richt, volgen, luister, opgeleverd*
5	*bezien, getrokken, behandeld, onderscheiden, luister, volgen, onderzocht, gekregen, geschieden, betekenen*

Model with window size 8. Table 17 presents the top ten changing verbs per run of model WIN 8[35]. In this model, 26 different verbs change with cosine distances between 0.904 (*bleven* in run) and 0.993 (*doorgaan* in run). The mean cosine distance is 0.940 with a standard deviation of 0.022. Nine of the 26 verbs change in three or more runs of the model: *doorgaan* (N=4, M=0.974, SD=0.014), *drijven* (N=4, M=0.930, SD=0.012), *eist* (N=4, M=0.948, SD=0.026), *bekeken* (N=3, M=0.939, SD=0.014), *belegd* (N=3, M=0.927, SD=0.012), *gevuld* (N=3, M=0.939, SD=0.012), *luister* (N=3, M=0.929, SD=0.011), *opgeroepen* (N=3, M=0.963, SD=0.024) and *voorkomt* (N=3, M=0.950, SD=0.030).

Model with window size 9. Using the WIN9 model[36], 29 verbs changed the most in the top tens for five runs, presented in Table 18. The verbs have a cosine distance between 0.917 (*heelt* in run 5) and 0.992 (*sluiten* in run 3), with a mean of 0.943 (SD=0.018). Seven of the 29 verbs change occur in the top tens of most changing verbs of three or more runs: *lachen* (N=4, M=0.958, SD=0.011), *leert* (N=4, M=0.939, SD=0.015), *sluiten* (N=4, M=0.951, SD=0.031), *bedacht* (N=3, M=0.953, SD=0.016), *bleven*(N=3, M=0.940, SD=0.010), *droeg* (N=3, M=0.927, SD=0.009) and *rusten* (N=3, M=0.935, SD=0.015).

Model with window size 10. Table 19 presents the 32 verbs from the top ten most changing verbs per run of the WIN10 model[37]. The cosine distances range from 0.901 (*keert* in run) to 0.995 (*bekeken* in run) (M=0.939, SD=0.023). Four of the 32 verbs change in three or more runs: *lijdt* (N=4, M=0.963, SD=0.031), *doorgaan* (N=3, M=0.949, SD=0.015), *vervolgt* (M=0.916, SD=0.009) and *volgt* (N=3, M=0.964, SD=0.021).

---

[35] Hyperparameters: skip gram, dimension size=100, window size=8, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[36] Hyperparameters: skip gram, dimension size=100, window size=9, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[37] Hyperparameters: skip gram, dimension size=100, window size=10, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

Table 17

The ten most changing verbs per run for the WIN8 model

Run	Ten most changing verbs
1	*doorgaan, vertoonde, ontsnapt, ophouden, ingeschakeld, bedacht, gesproken, gebonden, voorkomt, belegd*
2	*doorgaan, voorkomt, gelopen, bekeken, drijven, plaatst, ontsnapt, gold, eist, bleven*
3	*opgeroepen, eist, gelopen, geschonden, gevuld, belegd, verminderen, drijven, luidde, luister*
4	*doorgaan, voorkomt, opgeroepen, eist, drijven, gevuld, geschonden, belegd, bekeken, luister*
5	*doorgaan, verkeert, eist, bekeken, luister, opgeroepen, schaken, gevuld, hielden, drijven*

Table 18

The ten most changing verbs per run for the WIN9 model

Run	Ten most changing verbs
1	*gewerkt, sluiten, lachen, leert, kiest, droeg, uitoefenen, geschonden, getrokken, verdwijnen*
2	*leen, verbieden, verschijnen, leert, eist, besteden, voorspellen, voorkomt, heelt, verleent*
3	*sluiten, leen, bedacht, lachen, bleven, rusten, maak, leert, verrichten, droeg*
4	*lachen, bedacht, bleven, eist, verbieden, sluiten, steekt, leert, droeg, rusten*
5	*wachten, lachen, rusten, bedacht, bleven, gelopen, scoorde, inzien, sluiten, heelt*

Table 19

The ten most changing verbs per run for the WIN10 model

Run	Ten most changing verbs
1	*volgt, verklaart, veroorzaken, gebracht, rijdt, wint, gewenst, vervolgt, roepen, keert*
2	*brak, lijdt, doorgaan, veroorzaken, roepen, rijdt, vervolgt, gelezen, keert, draait*
3	*bekeken, volgt doorgaan, eist, lijdt, lachen, gelaten, zingen, luister, bedacht*
4	*brak, doorgaan, rusten, volgt, gewenst, verwerken, lijdt, verkeert, vervolgt, bekeken*
5	*lijdt, wekken, gelaten, sparen, schrijft, gehandeld, verkeert, gelopen, noemt, eindigt*

Model with dimension size 50. Using model DIM50[38], 23 different verbs occur in the top tens of the five runs of this model, presented in Table 20. The word with the highest cosine distance is *eist* in run 2 (0.996) and the word with the lowest cosine distance is *onderscheiden* in run 1 (0.896). The mean cosine distance of the verbs is 0.934 with a standard deviation of 0.027. Eight of the 23 verbs change in three or more runs of this model: *volgen* (N=5, M=0.942, SD=0.032), *eist* (N=4, M=0.946, SD=0.034), *gelopen* (N=4, M=0.961, SD=0.023), *toont* (N=4, M=0.956, SD=0.017), *uitkomen* (N=4, M=0.915, SD=0.009), *afgebroken* (N=3, M=0.932, SD=0.009), *onderscheiden* (N=3, M=0.932, SD=0.051) and *verklaart* (N=3, M=0.915, SD=0.010).

Model with dimension size 200. 27 unique verbs change when using the DIM200 model[39], shown in Table 21, with a mean cosine distance of 0.904 (SD=0.023). The cosine distances range from 0.868 (*loopt* in run 1) to 0.966 (*gebleken* in run 1). Seven verbs change in three or more runs: *loopt* (N=5, M=0.911, SD=0.030), *droegen* (N=4, M=0.912, SD=0.024), *opnemen* (N=4, M=0.919, SD=0.037), *slaagde* (N=4, M=0.890, SD=0.003), *verenigen* (N=4, M=0.926, SD=0.005), *gebonden* (N=3, M=0.0889, SD=0.014) and *gekregen* (N=3, M=0.914, SD=0.027).

---

[38] Hyperparameters: skip gram, dimension size=50, window size=5, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[39] Hyperparameters: skip gram, dimension size=200, window size=5, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

Table 20

The ten most changing verbs per run for the DIM50 model

Run	Ten most changing verbs
1	*kiest, gelopen, afgebroken, beslist, verklaart, dalen, bood, bleken, volgen, onderscheiden*
2	*eist, gelopen, toont, volgen, afgebroken, onderscheiden, slaat, kiest, meldde, uitkomen*
3	*volgen, toont, bleven, gelopen, verklaart, afgebroken, eist, geregeld, uitkomen, vertoont*
4	*onderscheiden, toont, slaat, eist, gebonden, ontbreken, volgen, uitkomen, zochten, gestaan*
5	*ontbreken, gelopen, vertoont, toont, volgen, eist, uitkomen, verklaart, gebonden, voorgeschreven*

Table 21

The ten most changing verbs per run for the DIM200 model

Run	Ten most changing verbs
1	*gebleken, verenigen, springen, ontbreekt, schieten, wint, vertoonde, ontworpen, opnemen, loopt*
2	*verenigen, droegen, vasthouden, bepalen, vertoonde, kocht, geprobeerd, loopt, schieten, slaagde*
3	*loopt, gekregen, droegen, opnemen, verenigen, bedoeld, kloppen, gebonden, slaagde, ondergaan*
4	*opnemen, verenigen, loopt, rusten, luidde, slaagde, verstaan, gekregen, droegen, gebonden*
5	*loopt, gekregen, opnemen, droegen, rusten, gedwongen, gebonden, volgen, draait, slaagde*

Model with dimension size 300. Using model DIM300[40], the top tens of the five runs contain 41 different verbs, see Table 22. The fifty verbs change with a mean cosine distance of 0.877 (*SD*=0.014), ranging from 0.858 (*spraken* in run 2) to 0.931 (*verwijderd* in run 4). Of the 41 unique verbs, only *verwijderd* (*N*=3, *M*=0.887, *SD*=0.038) occurs in the top tens of three or more runs.

Model with dimension size 400. As Table 23 indicates, 29 different verbs change in the top tens of the five runs of model DIM400[41]. The verbs change with cosine distances between 0.845 (*geleverd* in run 1) and 0.906 (*vertrouwde* in run 5), with a mean cosine distance of 0.868 (*SD*=0.015). Six of the 29 verbs occur in three or more top tens of the five runs: *luister* (*N*=5, *M*=0.871, *SD*=0.016), *verliest* (*N*=4, *M*=0.0881, *SD*=0.019), *afgebroken* (*N*=3, *M*=0.866, *SD*=0008), *afgewezen* (*N*=3, *M*=0.865, *SD*=0.019), *geleverd* (*N*=3, *M*=0.856, *SD*=0.012) and *vertrouwde* (*N*=3, *M*=0.892, *SD*=0.012).

Model with dimension size 500. Table 24 presents the ten most changing verbs per run of the DIM500 model[42]. The verbs change with cosine distances ranging from 0.826 (*schiet* in run 1) to 0.924 (*kijkt* in run 1), with a mean cosine distance of 0.855 (*SD*=0.022). The top tens contain 30 different verbs, of which five change most in three or more of the runs: *kijkt* (*N*=5, *M*=0.892, *SD*=0.026), *aangetoond* (*N*=4, *M*=0.849, *SD*=0.009), *geleverd* (*N*)=3, *M*=0.854, *SD*=0.008), *heelt* (*N*=3, *M*=0.856, *SD*=0.027) and *herhaalde* (*N*=3, *M*=0.848, *SD*=0.013).

Model with dimension size 600. The ten most changing verbs per run of the DIM600 model[43], are presented in Table 25. The verbs change with a mean cosine distance of 0.836 (*SD*=0.011), ranging from 0.820 (*ophouden* in run 1) to 0.869 (*verleent* in run 5). 35 different verbs change in these runs, of which two change in three or more: *gelopen* (*N*=5, *M*=0.847, *SD*=0.013) and *luister* (*N*=4, *M*=0.844, *SD*=0.016).

[40] Hyperparameters: skip gram, dimension size=300, window size=5, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[41] Hyperparameters: skip gram, dimension size=400, window size=5, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[42] Hyperparameters: skip gram, dimension size=500, window size=5, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

[43] Hyperparameters: skip gram, dimension size=500, window size=5, min count=100, negative sampling words=5, iterations=5, negative sampling rate=10e-3

Table 22

The ten most changing verbs per run for the DIM300 model

Run	Ten most changing verbs
1	*meldde, omschreven, eist, vullen, verliest, stoppen, verdwenen, probeert, geuit, keert*
2	*leidde, slaagde, rusten, luister, omschreven, behoeven, opgeroepen, probeert, verliest, spraken*
3	*zochten, mogen, droeg, wenste, bleken, ingeschakeld, vroegen, verlopen, voorgeschreven, verwijderd*
4	*vroegen, zochten, vond, ontmoeten, opgebouwd, krijg, verwijderd, heelt, neem, veroverde*
5	*verwijderd, aangetoond, bedoelt, opgebouw, stelt, behandeld, voorgeschreven, bewaard, horen, kijkt*

Table 23

The ten most changing verbs per run for the DIM400 model

Run	Ten most changing verbs
1	*gevolgd, rijdt, geniet, verliest, verenigen, gelopen, luister, bleken, afgewezen, geleverd*
2	*luister, verliest, bewijst, bedacht, afgewezen, opgeroepen, bezorgd, geleverd, gevolgd, durft*
3	*verliest, rusten, vertrouwde, luister, afgewezen, bewijst, aanvaarden, afgebroken, trekt, verschijnen*
4	*verliest, vertrouwde, droegen, luister, afgebroken, vluchten, opgegeven, uitgeroepen, verwijderd, werkt*
5	*vertrouwde, rusten, meegenomen, geleverd, aanvaarden, luister, vluchten, erkent, afgebroken, opgegeven*

Table 24

The ten most changing verbs per run for the DIM500 model

Run	Ten most changing verbs
1	*kijkt, zond, trouwen, heelt, droeg, herhaalde, overgedragen, wenste, overwegen, schiet*
2	*kijkt, zond, verliest, herhaalde, eist, aangetoond, gekregen, overwegen, raakt, gezocht*
3	*eist, kijkt, heelt, verliest, herhaalde, krijg, aangetoond, afgebroken, geleverd, verkregen*
4	*kijkt, vertrouwde, aangetoond, afgebroken, geleverd, werkt, gevochten, heelt, aanvaarden, eet*
5	*vertrouwde, ingaan, geleverd, kijkt, aanvaarden, bekeken, slaat, betekenen, kocht, aangetoond*

Table 25

The ten most changing verbs per run for the DIM600 model

Run	Ten most changing verbs
1	*wenste, luister, gelopen, gedwongen, overwegen, ingrijpen, droegen, geprobeerd, starten, ophouden*
2	*wenste, betaalde, luister, koos, binnengekomen, verricht, eindigde, uitkomen, gelopen, gekend*
3	*luister, gelopen, staken, verwerven, kijkt, gehandeld, denk, meende, beantwoorden, voorgeschreven*
4	*gelopen, klinkt, verleent, verwerven, staken, trekt, meegenomen, zendt, luister, beantwoorden*
5	*verleent, gelopen, koos, verricht, hoorde, trekt, verliest, afgebroken, leidde, gehoord*

Changing verbs. In total, 268 verbs change most in the 80 runs of the models (16 models with each 5 runs). Figure 3 presents the ranking of verbs with the amount of runs a verb changed in. The verb with the ranking number of 1 is the verb that changes in most runs of the models, in this case *luister*, compared to the other verbs. The verb with the ranking number of 2 is the verb that changes in most runs after *luister*, in this case *eist*. The verb with the ranking number of 268 is the verb that changes in fewest runs, in this case in only one.

109 of the 268 verbs change in only one run, from rank 160 to 268. These 109 verbs are ranked alphabetically, just as the other verbs that have changed in an equal amount of runs. *Behoeven*, changing in one run, thus has the ranking number of 160, while *zwemmen*, also changing in one run, has the ranking number of 268. 54 verbs change in only two runs, from rank 106 to 159.

Eight of the 268 verbs change in ten or more runs, these verbs are annotated in Figure 3. The verb form *luister* changed in 25 runs with a mean cosine distance of 0.903, and is therefore the verb most often changed. *Eist* changed in 17 runs with a mean cosine distance of 0.927, *gelopen* in 16 runs (M=0.912), *volgen* in 14 runs (M=0.938), *bedacht* in 12 runs (M=0.922), *rusten* in 12 runs (M=0.913), *onderscheiden* in 11 runs (M=0.937) and *zond* in 10 runs (M=0.909).

The figure that results from running bokehfigure.py presents an interactive alternative to Figure 3, with all 268 verbs annotated. When hoovering over the different data points, the corresponding verbs are shown with the amount of runs they changed in. The figure also has the option of zooming in on specific data points.

Stability. The stability of the model is measured by the amount of different verbs that change in the five top tens per model (the less the more stable), by the amount of verbs that change in three or more runs of the model (the more the more stable) and the standard deviation of the cosine distances of these verbs (the smaller the SD's the more stable).

Windows size. When changing the window size of the word2vec model from 5 to 1-10, the amount of verbs that change the most in the five runs of each model shifts from 27 in the default model to between 23, with window size set to 2, and 38, with window size set to 4. The models that have less or the same amount of verbs changing in the five top tens are the models with window size set to 1 (25 different verbs), 2 (23 verbs), 7 (26 verbs) and 8 (26 verbs).

Furthermore, 7 verbs change in three or more runs in the default model. In the models

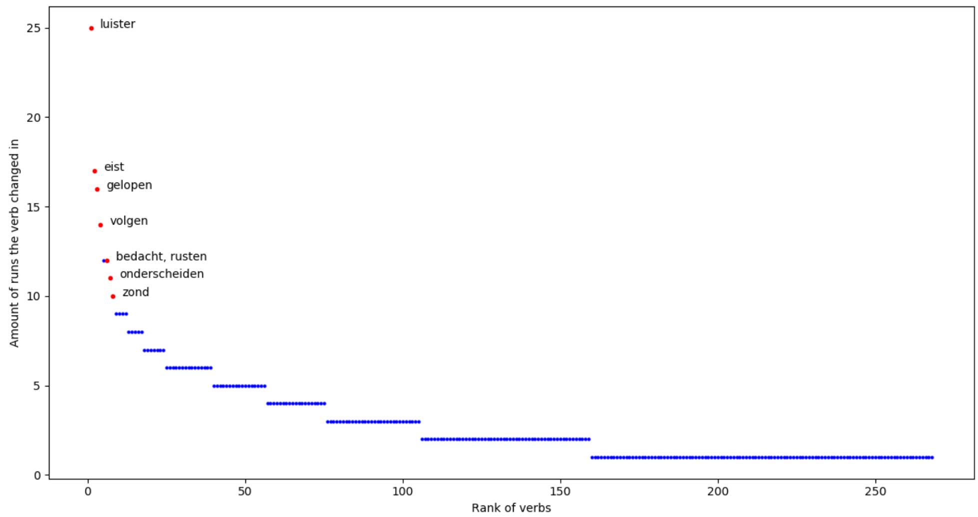


Figure 3. The ranking of verbs with the amount of runs it changed in (e.g. the verb on rank 1 is *luister* and changes in 25 runs).

with changing window sizes, between 3 and 9 verbs change in three or more runs. The models with 7 or more verbs changing, and thus possibly as stable, or more stable than, the default model are the models with window size set to 1 (8 verbs), 2 (8 verbs), 6 (7 verbs), 7 (7 verbs), 8 (9 verbs) and 9 (7 verbs).

Finally, when looking at the standard deviations of the verbs that change in three or more runs of a particular model, the standard deviations of the default model lie in between 0.011 (*behandelt*) and 0.039 (*luister*), with a mean standard deviation of 0.022. The models with 7 or more verbs changing in three or more runs of a model (WIN1, WIN2, WIN6, WIN7, WIN8, WIN9) have a mean standard deviation of those words between 0.015 (WIN9) and 0.020 (WIN2 and WIN6). Thus all models that have the same number of, or less than, verbs changing three or more times, all have a smaller mean standard deviation than the default model.

The models that perform as good as, or better than the default model, concerning stability, according to these numbers, are models WIN1, WIN2, WIN7 and WIN8.

Dimension size. When changing the dimension size of the default model from 100 to 50 and 200-600, between 23 (in DIM50) and 41 (in DIM300) different verbs change. Accordingly, in model DIM300, only eight verbs occur in two or more top tens of this model. DIM50 (23 different verbs) and DIM200 (27 different verbs) perform equal of better than the default model when looking at the amount of different verbs in the top tens of the models.

Regarding the amount of verbs that change in three or more runs of the corresponding model, the models with changing dimension sizes present between 1 (in DIM300) and 8 (in DIM50) verbs. The models with dimension size set to 50 and 200 have respectively 8 and 7 verbs changing as frequent as, or more frequent than, the default model.

The DIM200 model has a lower mean standard deviation than the default model, i.e. 0.020. The DIM50 model has a mean standard deviation of 0.023, which is only 0.001 higher than that of the default model.

Regarding the models with changing dimension size, the DIM50 and DIM200 clearly outperform the other models, and also slightly outperform the default model.

Frequent verbs and known changes

This subsection presents the changes that occur in the word forms of the fifteen earlier mentioned frequent verbs and verbs with known changes. I discuss the semantic change per verb lemma per model, by taking the mean of the cosine distances of the five runs of the corresponding verb forms and the standard deviation of those means that quantifies the amount of variation of the means of the forms[44]. When a verb lemma has a low standard deviation, the mean cosine distances of the verb forms are closely related. Appendix C shows the mean cosine distances of the five runs per verb form per lemma per model.

These mean cosine distances of the verb forms are used to measure the cosine distance of the corresponding lemma and the standard deviation [45]. To determine which verbs changed between 1945-1954 and 1985-1994, I analyse the mean cosine distance of the lemmas of all sixteen models, and the amount of models wherein a lemma was the most or least changing lemma.

To evaluate the stability, I discuss the standard deviation of the cosine distances across the five runs per model of the verb forms per lemma, for the twelve most frequent verbs[46]. I assess a model as more or less stable when the standard deviations of the verb forms, using

[44] For example, in the default model the verb forms of *zijn* have the following mean cosine distances: *ben*: 0.564, *bent*: 0.537, *is*: 0.379, etc. The standard deviation of these means is 0.105.

[45] Thus following the previous example, the cosine distance of the lemma *zijn* is 0.541 in the default model, and the standard deviation is 0.105.

[46] *zijn, worden, hebben, komen, gaan, maken, geven, doen, houden, zeggen, zien* and *krijgen*

that model, are lower or higher than in other models. These standard deviations are also presented in Appendix C.

Models. ***Default model.*** Table 26 presents the mean cosine distances per verb lemma, according to the default model, with the frequent verbs ranked by their frequencies. The twelve most frequent verb lemmas change with cosine distances between 0.522 (*worden*) and 0.700 (*zien*). For the five verbs with known changes, the cosine distances range from 0.642 (*doen*) to 0.764 (*bewegen*).

Doen also has the lowest standard deviation (0.036) of means of the corresponding verb forms, of the most frequent verbs, with all corresponding verb forms having a mean cosine distance between 0.598 (*doen*) and 0.689 (*deed*). *Zien* has the largest standard deviation, with the cosine distances of the verb forms ranging from 0.522 (*zag*) to 0.804 (*zagen*). However, *zagen* is a homonym, with a different unrelated meaning of 'to saw', which can cause the large cosine distance. *Krijgen* has the second highest standard deviation of the frequent verbs of 0.106. *Gekregen* has the lowest mean cosine distance over the five runs of the model (0.571), while *krijg* has the highest mean cosine distance of 0.854.

Table 26

The mean cosine distance per verb lemma for the default model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.541 (*SD*=0.105)	*houden*	0.684 (*SD*=0.086)
worden	0.522 (*SD*=0.093)	*zeggen*	0.638 (*SD*=0.043)
hebben	0.537 (*SD*=0.083)	*zien*	0.700 (*SD*=0.113)
komen	0.571 (*SD*=0.068)	*krijgen*	0.695 (*SD*=0.106)
gaan	0.583 (*SD*=0.059)	*bieden*	0.755 (*SD*=0.100)
maken	0.689 (*SD*=0.039)	*breken*	0.756 (*SD*=0.089)
geven	0.656 (*SD*=0.053)	*bewegen*	0.764 (*SD*=0.082)
doen	0.642 (*SD*=0.036)		

Model with window size 1. Using the WIN1 model, the cosine distances of the verb lemmas range from 0.562 (*zeggen*) to 0.693 (*breken*), see Table 27. For the twelve most frequent verbs, the cosine distances vary between 0.562 and 0.645 (*zijn*). For the verbs with known changes, the cosine distances vary between 0.594 (*doen*) and 0.693.

The frequent verb with the lowest standard deviation is *zijn*, with cosine similarities of its forms between 0.582 (*is*) and 0.696 (*zijn*). *Maken* has the highest standard deviation, of the frequent verbs, of 0.103, as *maakte* has a mean cosine distance of 0.733 and *maakte* of 0.483.

Model with window size 2. Table 28 shows the mean cosine distances per verb lemma for the WIN2 model. Using this model, the twelve most frequent verbs have cosine distances between 0.565 (*zijn*) and 0.667 (*houden*). The cosine distances of the verbs with known changes differ from 0.628 (*krijgen*) to 0.777 (*breken*).

Worden has the lowest standard deviation, 0.028, with its mean cosine distances between 0.570 (*werd*) and 0.642 (*wordt*). *Geven* has the highest standard deviation, 0.112. Its forms cosine distances range from 0.459 (*geven*) to 0.791 (*geef*).

Model with window size 3. The mean cosine distances of the fifteen verbs for the WIN3 model are presented in Table 29. *Krijgen* is the most changing verb of the most frequent verbs, with a mean cosine distance of 0.683, while *worden* is the least changing verb, with a mean cosine distance of 0.520. Of the verbs with known semantic changes, *breken*, changed the most with a cosine distance of 0.740, and *doen*, changed the least with a cosine distance of 0.602.

The verb with the smallest standard deviation is *gaan*, with the cosine distances of its verb forms ranging from 0.522 (*ga*) to 0.631 (*gaat*). The verb with the biggest standard deviation is *worden*, with cosine distances between 0.367 (*werd* and 0.627 (*word*).

Model with window size 4. As Table 30 shows, the most frequent lemmas change with a cosine distance between 0.559 (*hebben*) and 0.713 (*krijgen*) using the WIN4 model. The verbs with known changes have a cosine distance between 0.628 (*doen*) and 0.766 (*breken*).

Of the twelve most frequent verbs, *maken* has the lowest standard deviation of 0.033. Its verb forms change between 0.574 (*maken*) and 0.677 (*maakten*). *Zien* has the highest cosine distance, 0.117, with the cosine distance of its forms ranging from 0.516 (*zag*) to 0.797 (*zagen*). As *zagen*, we focus on the verb with the second highest cosine distance: *zijn*. The forms of the lemma *zijn* change between 0.443 (*is*) and 0.691 (*geweest*).

Table 27

The mean cosine distance per verb lemma for the WIN1 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.645 (*SD*=0.047)	*houden*	0.617 (*SD*=0.083)
worden	0.589 (*SD*=0.074)	*zeggen*	0.562 (*SD*=0.065)
hebben	0.604 (*SD*=0.077)	*zien*	0.638 (*SD*=0.061)
komen	0.616 (*SD*=0.051)	*krijgen*	0.614 (*SD*=0.067)
gaan	0.615 (*SD*=0.058)	*bieden*	0.692 (*SD*=0.035)
maken	0.605 (*SD*=0.103)	*breken*	0.693 (*SD*=0.120)
geven	0.636 (*SD*=0.095)	*bewegen*	0.652 (*SD*=0.028)
doen	0.594 (*SD*=0.054)		

Table 28

The mean cosine distance per verb lemma for the WIN2 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.565 (*SD*=0.058)	*houden*	0.667 (*SD*=0.088)
worden	0.604 (*SD*=0.028)	*zeggen*	0.613 (*SD*=0.076)
hebben	0.568 (*SD*=0.041)	*zien*	0.622 (*SD*=0.083)
komen	0.595 (*SD*=0.050)	*krijgen*	0.628 (*SD*=0.076)
gaan	0.596 (*SD*=0.057)	*bieden*	0.692 (*SD*=0.066)
maken	0.605 (*SD*=0.076)	*breken*	0.777 (*SD*=0.053)
geven	0.602 (*SD*=0.122)	*bewegen*	0.671 (*SD*=0.030)
doen	0.639 (*SD*=0.094)		

Table 29

The mean cosine distance per verb lemma for the WIN3 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.550 (*SD*=0.061)	*houden*	0.677 (*SD*=0.058)
worden	0.520 (*SD*=0.104)	*zeggen*	0.633 (*SD*=0.089)
hebben	0.556 (*SD*=0.071)	*zien*	0.642 (*SD*=0.081)
komen	0.589 (*SD*=0.065)	*krijgen*	0.683 (*SD*=0.054)
gaan	0.577 (*SD*=0.051)	*bieden*	0.708 (*SD*=0.128)
maken	0.598 (*SD*=0.092)	*breken*	0.740 (*SD*=0.063)
geven	0.633 (*SD*=0.079)	*bewegen*	0.730 (*SD*=0.055)
doen	0.602 (*SD*=0.055)		

Table 30

The mean cosine distance per verb lemma for the WIN4 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.569 (*SD*=0.082)	*houden*	0.667 (*SD*=0.066)
worden	0.572 (*SD*=0.073)	*zeggen*	0.637 (*SD*=0.065)
hebben	0.559 (*SD*=0.076)	*zien*	0.659 (*SD*=0.117)
komen	0.590 (*SD*=0.055)	*krijgen*	0.713 (*SD*=0.072)
gaan	0.608 (*SD*=0.080)	*bieden*	0.742 (*SD*=0.081)
maken	0.623 (*SD*=0.033)	*breken*	0.766 (*SD*=0.069)
geven	0.625 (*SD*=0.070)	*bewegen*	0.747 (*SD*=0.077)
doen	0.628 (*SD*=0.081)		

Model with window size 6. *Table* 31 shows the mean cosine distances of the fifteen verb lemmas when the word2vec model is run with window size set to 6. The frequent verb that changes most is *zien*, with a mean cosine distance of 0.700, and the verb that changes the least is *zijn*, with a mean cosine distance of 0.512. Of the verbs with known changes, *bewegen* changes the most (M=0.724), and *doen* changes the least (M=0.648).

When considering the standard deviations of the most frequent verbs, *zeggen* has the lowest (SD=0.015). Its verb forms change with a cosine distance between 0.576 (*zei*) and 0.621 (*zeg*). *Zijn* has the highest standard deviation (SD=0.118), with the cosine distances of its forms ranging from 0.307 (*is*) to 0.649 (*geweest*).

Model with window size 7. Using the WIN7 model (Table 32), *krijgen* is the most changing frequent verb with a cosine distance of 0.732 and *zijn* is the least changing frequent verb with a cosine distance of 0.496. Of the verbs with known changes, *doen* changes the least (M=0.666) and *breken* changes the most (M=0.795).

Of the twelve most frequent verbs, *zeggen* has the lowest standard deviation (SD=0.057). Its verb forms change with a cosine distance between 0.562 (*zegt*) and 0.687 (*zeggen*). *Worden* has the highest standard deviation (SD=0.116), with its forms changing with a cosine distance between 0.417 (*worden* and 0.689 (*word*).

Model with window size 8. The mean cosine distances of the fifteen verbs using the WIN8 model are presented in Table 33. *Houden* changes the most of the most frequent verbs, with a cosine distance of 0.697. *Zijn* changes the least, with a cosine distance of 0.552. Of the verbs with known changes, *bieden* has the biggest change (M=0.774) and *doen* has the smallest change (M=0.627).

Of the twelve most frequent verbs, *geven* has the lowest standard deviation (SD=0.046). The cosine distances of the forms of this lemma are between 0.636 (*geef*) and 0.768 (*gegeven*). *Hebben* and *zien* both have the highest standard deviation (SD=0.118). The cosine distances of *hebben* lay between 0.440 (*heb*) and 0.710 (*gehad*). The cosine distances of *zien* are between 0.598 (*zien* and *ziet*) and 0.903 (*zagen*).

Table 31

The mean cosine distance per verb lemma for the WIN6 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.512 (*SD*=0.118)	*houden*	0.682 (*SD*=0.044)
worden	0.524 (*SD*=0.102)	*zeggen*	0.595 (*SD*=0.015)
hebben	0.514 (*SD*=0.081)	*zien*	0.700 (*SD*=0.107)
komen	0.626 (*SD*=0.093)	*krijgen*	0.681 (*SD*=0.050)
gaan	0.613 (*SD*=0.045)	*bieden*	0.712 (*SD*=0.129)
maken	0.613 (*SD*=0.100)	*breken*	0.719 (*SD*=0.041)
geven	0.645 (*SD*=0.103)	*bewegen*	0.724 (*SD*=0.041)
doen	0.648 (*SD*=0.086)		

Table 32

The mean cosine distance per verb lemma for the WIN7 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.496 (*SD*=0.112)	*houden*	0.639 (*SD*=0.102)
worden	0.530 (*SD*=0.116)	*zeggen*	0.616 (*SD*=0.057)
hebben	0.570 (*SD*=0.112)	*zien*	0.732 (*SD*=0.083)
komen	0.612 (*SD*=0.079)	*krijgen*	0.735 (*SD*=0.074)
gaan	0.649 (*SD*=0.078)	*bieden*	0.742 (*SD*=0.090)
maken	0.666 (*SD*=0.079)	*breken*	0.795 (*SD*=0.118)
geven	0.643 (*SD*=0.080)	*bewegen*	0.785 (*SD*=0.054)
doen	0.666 (*SD*=0.087)		

Table 33

The mean cosine distance per verb lemma for the WIN8 model

Verb lemma	Mean cosine distance		
zijn	0.522 (*SD*=0.097)	*houden*	0.697 (*SD*=0.087)
worden	0.529 (*SD*=0.089)	*zeggen*	0.590 (*SD*=0.070)
hebben	0.562 (*SD*=0.118)	*zien*	0.678 (*SD*=0.118)
komen	0.607 (*SD*=0.103)	*krijgen*	0.696 (*SD*=0.073)
gaan	0.611 (*SD*=0.091)	*bieden*	0.774 (*SD*=0.138)
maken	0.672 (*SD*=0.069)	*breken*	0.772 (*SD*=0.131)
geven	0.683 (*SD*=0.046)	*bewegen*	0.722 (*SD*=0.033)
doen	0.627 (*SD*=0.077)		

Model with window size 9. Table 34 presents the mean cosine distances of the fifteen verb lemmas using the WIN9 model. The verb lemma that changes the most of the frequent verbs is *zien*, with a cosine distance of 0.731. The lemma that changes the least of the frequent verbs is *zijn* (*M*=0.519). *Bieden* changes the most of the verbs with known changes, with a cosine distance of 0.791. *Doen* changes the least, with a cosine distance of 0.698.

The frequent verb with the lowest standard deviation is *krijgen* (*SD*=0.015). Its forms have a cosine distance ranging from 0.632 (*kreeg*) to 0.673 (*krijg*). The verb with the highest standard deviation is *maken* (*SD*=0.105), with the cosine distances of its forms ranging from 0.565 (*gemaakt*) to 0.876 (*maak*).

Model with window size 10. The mean cosine distances of the fifteen verbs using the WIN10 model are presented in Table 35. As in some previous models, *zien* is the most changing frequent verb, with a cosine distance of 0.717. *Zijn* is the least changing frequent verb, also in accordance with previous models, with a cosine distance of 0.516. The cosine distances of the verbs with known changes range from 0.605 (*doen*) to *breken* (0.799).

The standard deviations of the most frequent verb lemmas range from 0.056 (*krijgen*) to 0.121 (*hebben*). The forms of *krijgen* have cosine distances between 0.578 (*kreeg* and 0.726 (*krijg* and *krijgt*). The forms of *hebben* have cosine distances between 0.390 (*hebben*) and 0.667 (*gehad*).

Table 34

The mean cosine distance per verb lemma for the WIN9 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.519 (*SD*=0.102)	*houden*	0.663 (*SD*=0.051)
worden	0.562 (*SD*=0.091)	*zeggen*	0.589 (*SD*=0.067)
hebben	0.540 (*SD*=0.099)	*zien*	0.731 (*SD*=0.089)
komen	0.645 (*SD*=0.099)	*krijgen*	0.659 (*SD*=0.015)
gaan	0.663 (*SD*=0.089)	*bieden*	0.791 (*SD*=0.091)
maken	0.680 (*SD*=0.105)	*breken*	0.749 (*SD*=0.045)
geven	0.672 (*SD*=0.055)	*bewegen*	0.735 (*SD*=0.021)
doen	0.698 (*SD*=0.063)		

Table 35

The mean cosine distance per verb lemma for the WIN10 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.516 (*SD*=0.111)	*houden*	0.677 (*SD*=0.067)
worden	0.559 (*SD*=0.116)	*zeggen*	0.596 (*SD*=0.096)
hebben	0.542 (*SD*=0.121)	*zien*	0.717 (*SD*=0.093)
komen	0.604 (*SD*=0.085)	*krijgen*	0.676 (*SD*=0.056)
gaan	0.662 (*SD*=0.071)	*bieden*	0.776 (*SD*=0.108)
maken	0.708 (*SD*=0.058)	*breken*	0.799 (*SD*=0.068)
geven	0.652 (*SD*=0.098)	*bewegen*	0.794 (*SD*=0.072)
doen	0.605 (*SD*=0.098)		

Model with dimension size 50. Table 36 presents the mean cosine distances of the fifteen verb lemmas, using the DIM50 model. Of the twelve frequent verbs, *zijn* changes the least, with a mean cosine distance of 0.363, and *zien* changes the most, with a mean cosine distance of 0.632. Of the verbs with known changes, *doen* has the lowest cosine distance of 0.531 and *bewegen* has the highest cosine distance of 0.743.

Krijgen has the lowest standard deviation (*SD*=0.036). Its verb forms change with cosine distances ranging from 0.563 (*kreeg*) to 0.660 (*kregen*). *Hebben* is the frequent verb with the highest standard deviation (*SD*=0.150). Its verb forms have cosine distances between 0.272 (*heeft*) and 0.672 (*hebt*).

Model with dimension size 200. Results using the word2vec model with dimension size set to 200 are presented in Table 37. The least changing frequent verb lemma is *worden* with a mean cosine distance of 0.650. The most changing frequent verb lemma is *krijgen* with a cosine distance of 0.734. The verb with known changes change with a cosine distance between 0.685 (*doen*) and 0.780 (*breken*).

The standard deviations of the most frequent verbs range from 0.037 (*gaan*) to 0.085 (*krijgen*). The forms of *gaan* change between 0.615 (*ga* and 0.712 (*gegaan*). The forms corresponding to *krijgen* change between 0.642 (*kreeg*) and 0.861 (*gekregen*).

Model with dimension size 300. Table 38 presents the mean cosine distance of the frequent verbs and verbs with known changes using the DIM300 model. The cosine distances of the frequent verbs range from 0.696 (*zijn*) to 0.752 (*krijgen*). The cosine distances of the verbs with known changes range from 0.713 (*doen*) to 0.792 (*breken*).

The standard deviations of the frequent verbs range from 0.024 (*zijn*) to 0.050 (*hebben*), disregarding the standard deviation of *zien* as some of its forms are homonyms. The forms of the lemma *zijn* change between 0.665 (*is*) and 0.726 (*waren*). The forms of the lemma *hebben* change between 0.599 (*heb*) and 0.737 (*had*).

Table 36

The mean cosine distance per verb lemma for the DIM50 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.363 (*SD*=0.083)	*houden*	0.588 (*SD*=0.096)
worden	0.385 (*SD*=0.110)	*zeggen*	0.522 (*SD*=0.163)
hebben	0.413 (*SD*=0.150)	*zien*	0.632 (*SD*=0.118)
komen	0.482 (*SD*=0.102)	*krijgen*	0.599 (*SD*=0.036)
gaan	0.510 (*SD*=0.141)	*bieden*	0.668 (*SD*=0.207)
maken	0.519 (*SD*=0.097)	*breken*	0.663 (*SD*=0.196)
geven	0.535 (*SD*=0.089)	*bewegen*	0.743 (*SD*=0.118)
doen	0.531 (*SD*=0.103)		

Table 37

The mean cosine distance per verb lemma for the DIM200 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.669 (*SD*=0.044)	*houden*	0.709 (*SD*=0.077)
worden	0.650 (*SD*=0.048)	*zeggen*	0.708 (*SD*=0.049)
hebben	0.669 (*SD*=0.052)	*zien*	0.729 (*SD*=0.074)
komen	0.665 (*SD*=0.056)	*krijgen*	0.734 (*SD*=0.085)
gaan	0.668 (*SD*=0.037)	*bieden*	0.750 (*SD*=0.067)
maken	0.677 (*SD*=0.061)	*breken*	0.780 (*SD*=0.071)
geven	0.691 (*SD*=0.041)	*bewegen*	0.754 (*SD*=0.016)
doen	0.685 (*SD*=0.038)		

Table 38

The mean cosine distance per verb lemma for the DIM300 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.696 (*SD*=0.024)	*houden*	0.723 (*SD*=0.027)
worden	0.707 (*SD*=0.046)	*zeggen*	0.723 (*SD*=0.045)
hebben	0.678 (*SD*=0.050)	*zien*	0.740 (*SD*=0.051)
komen	0.708 (*SD*=0.025)	*krijgen*	0.752 (*SD*=0.049)
gaan	0.700 (*SD*=0.033)	*bieden*	0.763 (*SD*=0.033)
maken	0.723 (*SD*=0.032)	*breken*	0.792 (*SD*=0.058)
geven	0.703 (*SD*=0.038)	*bewegen*	0.748 (*SD*=0.066)
doen	0.713 (*SD*=0.026)		

Model with dimension size 400. Table 39 presents the mean cosine distance of the verb lemmas using the DIM500 model. The most changing frequent verbs is *zien* with a cosine distance of 0.749 and the least changing frequent verb is *hebben.* The most changing verb with known changes is again *breken* with a cosine distance of 0.798 and the least changing verb with known changes is again *doen*, with a cosine distance of 0.710.

The standard deviations of the frequent verbs range from 0.028 (*zijn*) to 0.054 (*maken*). The form of *zijn* with the smallest cosine distance is *is* (*M*=0.682) and with the biggest cosine distance is *was* (*M*=0.756). The form of *maken* with the smallest cosine distance is *gemaakt* (*M*=0.644) and with the biggest cosine distance is *maak* (*M*=0.809).

Model with dimension size 500. Table 40 gives the cosine distances of the fifteen verbs using the DIM500 model. The least changing verb is *doen* (*M*=0.678), which is both one of the most frequent verbs and one of the verbs with known changes. The most changing frequent verb is *krijgen*, with a mean cosine distance of 0.760, and the most changing verb with known changes is *breken*, with a cosine distance of 0.787.

The standard deviations of the frequent verbs range from 0.010 (*geven*) to 0.048 (*maken*). The form of *geven* that changes least *geef*, having a mean cosine distance 0.688, and the form that changes most is *geven*, with a mean cosine distance of 0.714. Of the lemma *maken*, *gemaakt* changes least, with a cosine distance of 0.664, and *maak* changes most, with a cosine distance of 0.806.

Model with dimension size 600. Table 41 presents the mean cosine distances of the verb lemmas of the fifteen verbs, using the word2vec model with dimension size set to 600. According to these results, *komen* is the frequent verb lemma with the smallest mean cosine distance (M=0.692) and *krijgen* is the frequent verb lemma with the largest mean cosine distance (M=0.741). Regarding the verbs with known changes, *doen* has the smallest change (M=0.693) and *breken* has the largest change (M=0.766).

Concerning the standard deviations of the frequent verbs, *doen* has the smallest standard deviation (SD=0.014). Its forms change with a standard deviation between 0.676 (*deed* and 0.712 (*doe*). *Hebben* has the highest standard deviation (SD=0.048), with its forms having a standard deviation between 0.662 (*heeft*) and 0.786 (*gehad*).

Table 39

The mean cosine distance per verb lemma for the DIM400 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.720 (SD=0.028)	*houden*	0.722 (SD=0.034)
worden	0.730 (SD=0.031)	*zeggen*	0.736 (SD=0.039)
hebben	0.698 (SD=0.053)	*zien*	0.749 (SD=0.049)
komen	0.703 (SD=0.047)	*krijgen*	0.748 (SD=0.042)
gaan	0.740 (SD=0.049)	*bieden*	0.748 (SD=0.029)
maken	0.735 (SD=0.054)	*breken*	0.798 (SD=0.050)
geven	0.719 (SD=0.041)	*bewegen*	0.728 (SD=0.022)
doen	0.710 (SD=0.033)		

Table 40

The mean cosine distance per verb lemma for the DIM500 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.703 (*SD*=0.031)	*houden*	0.730 (*SD*=0.025)
worden	0.729 (*SD*=0.027)	*zeggen*	0.742 (*SD*=0.028)
hebben	0.700 (*SD*=0.047)	*zien*	0.736 (*SD*=0.039)
komen	0.702 (*SD*=0.044)	*krijgen*	0.760 (*SD*=0.045)
gaan	0.707 (*SD*=0.021)	*bieden*	0.741 (*SD*=0.040)
maken	0.722 (*SD*=0.048)	*breken*	0.787 (*SD*=0.062)
geven	0.704 (*SD*=0.010)	*bewegen*	0.747 (*SD*=0.020)
doen	0.678 (*SD*=0.030)		

Table 41

The mean cosine distance per verb lemma for the DIM600 model

Verb lemma	Mean cosine distance	Verb lemma	Mean cosine distance
zijn	0.705 (*SD*=0.029)	*houden*	0.711 (*SD*=0.045)
worden	0.700 (*SD*=0.040)	*zeggen*	0.729 (*SD*=0.025)
hebben	0.706 (*SD*=0.048)	*zien*	0.718 (*SD*=0.060)
komen	0.692 (*SD*=0.025)	*krijgen*	0.741 (*SD*=0.043)
gaan	0.701 (*SD*=0.021)	*bieden*	0.740 (*SD*=0.035)
maken	0.704 (*SD*=0.037)	*breken*	0.766 (*SD*=0.060)
geven	0.702 (*SD*=0.020)	*bewegen*	0.752 (*SD*=0.013)
doen	0.693 (*SD*=0.014)		

Changing verbs. ***Frequent verbs.*** Table 42 presents the mean cosine distances of the most frequent verb lemmas for all models, ranked by value of cosine distance. *Zijn* has the lowest mean cosine distance (M=0.581) and is also the least changing lemma in eight of the sixteen models, thus is, according to the word2vec models, probably the least changing frequent verb. However, it is at the same time the most changing verb in the WIN1 model.

The verb lemmas with the highest mean cosine distances, and thus probably the most changing frequent verbs, are *zien* and *krijgen*, both with cosine distances of 0.695. Together, the verbs are the most changing verbs in thirteen of the sixteen models, as *zien* is the verb with the highest cosine distance in six models, and *krijgen* in seven models. The cosine distances of *zien* and *krijgen* are significantly higher than the cosine distance of *zijn*, respectively $t(30)=-4.21$, $p<.001$ and $t(30)=-4.08$, $p<.001$.

However, the cosine distances of the most changing verb *krijgen* are significantly lower, and thus also of *zien*, than the mean cosine distances per model of the discovered changes, $t(30)=-14.04$, $p<.001$.

Verbs with known changes. Table 43 presents the mean cosine distances per verb lemma for all models for the verbs with known changes. The verb with the lowest mean cosine distance is *doen* (M=0.647). Likewise, *doen* is the verb that changes the least, compared to the other verbs with known changes, in fifteen of the sixteen models.

The verb with the highest mean cosine distance of all verbs with known changes, and thus probably the most changing verb, is *breken* (M=0.723). *Breken* is also the verb that is the most changing verb in most of the models, namely in eleven of the sixteen. The mean cosine distance of *breken* is significantly higher than the mean cosine distance of *doen*, $t(30)=2.73$, $p=.013$.

The most changing verb, of the verbs with known changes, does not have a significantly higher cosine distance than the most changing frequent verb, $t(30)=1.19$, $p=.246$. The cosine distance of *breken* still has a significantly lower cosine distance than verbs with the automatically discovered changes, $t=-5.93$, $p<.001$.

Table 42

The mean cosine distance per verb lemma for all models for the most frequent verbs

Lemma	Mean cosine distance	Lemma	Mean cosine distance
zijn	0.581	*doen*	0.647
worden	0.588	*geven*	0.656
hebben	0.589	*maken*	0.659
komen	0.625	*houden*	0.678
gaan	0.638	*zien*	0.695
zeggen	0.639	*krijgen*	0.695

Table 43

The mean cosine distance per verb lemma for all models for the verbs with known changes

Lemma	Mean cosine distance	Lemma	Mean cosine distance
doen	0.647	*bewegen*	0.711
krijgen	0.695	*breken*	0.723
bieden	0.711		

Stability. To study the stability of the sixteen models, focusing on the stability of cosine distance of frequent verbs, I discuss the standard deviations of the verb forms across the five runs of a model. The twelve frequent verbs consist of 75 verb forms in total. Each verb form has its own standard deviation per model, e.g. the verb form *ben* has in the default model a standard deviation of 0.035. By taking the mean of the standard deviations of the different verb forms per lemma, I create a standard deviation per verb lemma per model. This way, I do not have to compare 75 verb forms per model, but twelve verb forms per model. Using the default model, the verb lemmas have the following mean standard deviations: *zijn*: *SD*=0.034, *worden*: *SD*=0.020, *hebben*: *SD*=0.028, *komen*: *SD*=0.022, *gaan*: *SD*=0.032, *maken*: *SD*=0.029, *geven*: *SD*=0.026, *doen*: *SD*=0.027, *houden*: *SD*=0.033, *zeggen*: *SD*=0.032, *zien*: *SD*=0.030 and *krijgen*: *SD*=0.033. The mean standard deviation of these verb lemmas is 0.029. Appendix D presents the standard deviations of the other models per verb lemma per model.

Window size. When comparing the models with the window size set to different values, the WIN1 model has a, not significantly, higher mean standard deviation (M=0.033), t(22)=1.58, p=.131, and has lower cosine distances than the default model for five verb lemmas. The WIN2 model has the same mean standard deviation as the WIN1 model and has lower cosine distances than the default model for only two verbs. The WIN3, WIN4, WIN6, WIN7, WIN9 and WIN10 have higher mean standard deviations, between 0.031 and 0.041 and have lower standard deviations for between one and five verb lemmas. The only model with a lower mean standard deviation is the WIN8 model, with a mean standard deviation of 0.028. However this difference is not significant, t(22)=-0.25, p=.807. The WIN8 model has a lower standard deviation of five of the verb lemmas, a higher standard deviation of five other verb lemmas, and the same standard deviation as the other two verb lemma, than/as the default model.

The WIN8 model is the only model that possible performs more stable than the default model.

Dimension size. When comparing the models with the dimension size set to different values, DIM50, DIM200 and DIM300 have higher mean standard deviations, ranging from 0.032 and 0.035, than the default model. The mean standard deviations of the DIM50 and DIM200 model are not significantly higher, respectively t(22)=1.44, p=.164 and t(22)=1.21, p=.242. The mean standard deviation of the DIM300 model is significantly higher, t(22)=2.86, p=.010. The DIM400 model has a, not significantly, lower mean standard deviation, t(22)=-0.63, p=.535. Both the DIM500 and DIM600 models have a significantly lower mean standard deviation, respectively 0.024, t(22)=-2.32, p=.030, and 0.022, t(22)=-4.22, p<.001.

The same models that score worse on mean standard deviation, DIM50, DIM200 and DIM300, also have higher standard deviations for most of the individual verb lemmas, with the standard deviations of between seven and ten verb lemmas higher than the standard deviations of the corresponding verb lemmas in the default model. The models with a lower mean standard deviation, DIM400, DIM500 and DIM600, have lower standard deviations for between six and ten verb lemmas than the corresponding lemmas in the default model. The DIM400 model has lower standard deviations than the default model for six verbs and higher standard deviations for the other six verbs. The DIM500 model has lower standard deviation for eight verbs, the same standard deviation for one verb, and higher standard deviations for three verbs. DIM600 has a lower standard deviation for all verb lemmas, except *geven* and *worden*, than the corresponding lemmas in the default model.

The DIM400, DIM500 and DIM600 models are the models that appear to perform more stable than the default model.

Conclusion

The results of this study on the use of the word2vec model using a Dutch historical newspaper corpus present answers to the following research questions:

1. Which changes does the model discover in verbs?
2. When is the model most stable regarding discovered changes?
3. How do frequent verbs change?
4. When is the model most stable regarding frequent verbs?
5. Does the model capture known historical semantic changes?

To answer the first question, the model detects 268 different verb forms that appear in the top tens of most changing verbs of all eighty runs. Eight of these verbs change in ten or more of the runs, and are thus most likely to have changed. These verbs and the amount of runs they changed in are: *luister* (N=25), *eist* (N=17), *gelopen* (N=16), *volgen* (N=14), *bedacht* (N=12), *rusten* (N=12), *onderscheiden* (N=11) and *zond* (N=10). These verbs change with a mean cosine distance between 0.903 and 0.938.

These most changing verbs, detected by the word2vec model, are activities, accomplishments, achievements and states. *Luisteren*, *lopen*, *volgen* and *rusten* can be activities or accomplishments, dependent on the context. *Eisen* and *bedenken* are accomplishments. *Onderscheiden* is a state, and *zenden* is an achievement.

The results for the automatically detected changes using the word2vec model are most stable (1) when few different verbs appear in the five different top tens of a particular model, as there are five runs per model, (2) when many of these words appear in more than halve of the top tens, and (3) when the standard deviation of the cosine distances of a verb, that appears in more than halve of the top tens, are relatively low. Following this, the models that are more stable than the default model are for the models with changed window sizes the WIN1, WIN2, WIN7 and WIN8 models. For the models with changed dimension sizes, the DIM50 and DIM200 models are more stable than the default model.

When comparing the results of the models regarding the automatically detected changes with changed window sizes and changed dimension sizes, there are no significant differences between the models. The WIN models have between 23 and 26 different verbs, the DIM model

23 and 27, and thus have the same mean amount of different verbs, $t(4)$=0.0, p=1.0. The models have no significant difference in how many verbs change in three or more runs, $t(4)$=0.45, p=0.71, and no significant difference in there standard deviations, $t(4)$=-1.84, p=0.26.

Furthermore, the different WIN models present changes in the top tens of all runs with a mean cosine distance of 0.923 and a standard deviation of 0.020. The different DIM models present changes with a mean cosine distance of 0.879 and a standard deviation of 0.035, with higher values of dimension size resulting in lower cosine distances. As the standard deviation of the DIM models is higher than the standard deviation of the WIN models, the parameter setting of the dimension size is likely to have more influence on stability than the parameter setting of the window size.

Concerning the third question, the most frequent verb lemmas all change less than the automatically detected verb forms. The least changing, and most frequent, lemma is *zijn*, a state, with a mean cosine distance of 0.581 across all sixteen model. The most changing lemmas are *krijgen*, an achievement, and *zien*, an activity or accomplishment, both with mean cosine distances of 0.695. These two verbs are also the least frequent verbs of the twelve most frequent verbs. This, together with the fact that the frequent verbs all change less than the automatically detected changes, supports Hamilton's law of conformity (Hamilton et al., 2016b).

The results for the changes in frequent verbs are most stable (1) when the standard deviations of the verb forms are relatively low and (2) when many lemmas have a lower standard deviation than the default model. These indicators are intertwined. I studied 75 verb forms that belong to twelve verb lemmas. The mean of the standard deviations of the verb forms belonging to a lemma determine the standard deviation of that verb lemma. When combining the twelve standard deviations of the twelve frequent verbs into one mean standard deviation, the model is also assigned a standard deviation. Regarding the models with changed window sizes, the WIN8 model outperforms the default model. Regarding the models with changed dimension sizes, the DIM400, DIM500 and DIM600 models outperform the default model.

Comparing the WIN models and DIM models shows that the best performing DIM models outperform the best performing WIN model on stability, as the DIM400, DIM500 and

DIM600 have lower standard deviations, and more lemmas with lower deviations, than the WIN8 model. Furthermore, the WIN models present changes of the most frequent verbs with a mean cosine distance of 0.619 and a standard deviation of 0.010. The DIM models present changes with a mean cosine distances of 0.677 and a standard deviation of 0.085, with higher values in dimension size resulting in higher cosine distances. As the standard deviation between DIM models is higher than the standard deviation between WIN model, it may be that changing the dimension size has more influence on stability than changing the window size, in case of semantic change in frequent verbs, and that setting the right parameter value for dimension size is more important than setting the right parameter value for window size.

To answer the last question, all sixteen models capture some changes in the five verbs with known changes, as they have mean cosine distances ranging from 0.647 to 0.723. However, all verb lemmas with known historical shifts change significantly less than the automatically discovered changes according to the word2vec models. *Doen*, an activity or accomplishment, is the least changing of these verbs, but also the most frequent. *Breken*, an achievement, is the most changing verb of the verbs with known changes. The cosine distance of *breken* is higher, but not significantly, than the cosine distance of *krijgen*, which is the most changing frequent verb, and lower than the cosine distances of the automatically discovered changes.

Discussion

This research presented eight different verbs that might have changed between 1945-1954 and 1985-1994 according to the word2vec model, evidence that supports the law of conformity Hamilton et al. (2016b) and evidence that hyperparameter setting has influence on the results and stability of the model, with the hyperparameter setting of dimension size having more influence on stability than the setting of window size.

Related work

The results of this study support the law of conformity by Hamilton et al. (2016b), as frequent words change more slowly. To investigate this, I followed part of the methodology and recommendations of Hamilton et al. (2016b) for constructing word vectors using SGNS, for aligning them over time, and for measuring semantic similarity using the cosine similarity. I also followed part of their methodology for automatically detecting shifts. Hamilton et al. (2016b) studies semantic changes using PPMI, SVD and SGNS models. I have not constructed models using PPMI or SVD, as the SGNS architecture performed best on discovering shifts from data. However, as I have not constructed models using PPMI or SVD, I have not tested if SGNS was also the best performing architecture for my dataset.

I have also not followed Hamilton et al. (2016b) in evaluating my top tens discovered shifts using existing literature or historical corpora to judge them as genuine, borderline or corpus artifacts. This is because I decided to not focus on quality in particular, as there is very little available literature on Dutch semantic changes or on how to judge semantic changes, other than reviewing dictionaries.

Lastly, Hamilton et al. (2016b) focused on both frequency and polysemy, and their influence on semantic change, while I only focused on frequency to make this study more feasible. Part of the reason for this is that lots of verbs are part of compound verbs. This makes the number of senses of a token very high, the Dutch verb *gaan* has for example 17 senses, but including the senses of its compound verbs, *gaan* has 97 senses.

In another study, Hamilton et al. (2016a) have shown two measures for semantic change: the global distance measure and the local neighbourhood measure. In my study, I have used the global distance measure, as that proved to be more sensitive to semantic change in verbs. I have not compared changes in verbs with changes in nouns, and I have not compared the global distance measure with the local neighbourhood measure, as I have not applied that second measure. I thus cannot confirm nor deny the results of Hamilton et al. (2016a).

The study of Haagsma and Nissim (2017) I used for the list of Dutch words with known historical changes. As there is not an enormous diachronic Dutch corpus available, they also constructed their own corpus using Dutch newspapers as they are best available online. In contrast with my corpus, there was only a time difference of 22 years between the oldest and newest text, and their subcorpora only consisted of between 4.3 to 35.6 million tokens. They briefly discuss the distinctive feature of compounding in Dutch. That has also come up in my study, as a change in the token *maak* can indicate a change in multiple lemmas containing **maken*.

The stability of word vectors is an often researched topic (Chugh et al., 2018; Hellrich & Hahn, 2016a, 2016b; Pierrejean & Tanguy, 2018a, 2018b; Tahmasebi, 2018). Stability refers in these studies to the consistency of the neighbourhood of words around a given target word. However, I have chosen to measure stability in a different manner, namely by measuring the consistency in cosine distances of a given target word. As this is not based on existing literature, it might not be a perfect measure, although it did produce clear results.

I followed part of the methodology of Pierrejean and Tanguy (2018a) by retraining a model with the exact same hyperparameters for five times, following the parameter settings of Pierrejean and Tanguy's study in setting the parameters in my study. I followed part of the methodology of Pierrejean and Tanguy (2018b) by constructing different models with one parameter value changed at a time. In this second study of Pierrejean and Tanguy, they measure the variation between the different models compared to the default model. I chose to combine these to studies, by constructing sixteen different models and retraining each five times. I then measured the stability per model and compared the stability of all models with the default model.

The study of Pierrejean and Tanguy (2018b) showed that even for models with the smallest variation, the average variation score is at least 0.3, meaning that by changing one parameter, 1 out of 3 neighbours of a given word is different from one model to another. My study shows similar results, as the ten most frequent verbs per model differ much between models, more than in the five runs of a particular model.

Furthermore, Pierrejean and Tanguy's study shows that changing the architecture or the corpus generates the most variation, and that changing the vector size and window size the least, respectively from 100 to 200 and from 5 to 6, causes the lowest variation. These results for vector and window size are partly consistent with the results of this present study when looking at the variation in each model, as the difference in variation is biggest between the default model and the models with maximum values. However, when changing the window size from 5 to 3 or 4, the differences in variation are as high as or higher than when changing from 5 to 10.

Limitations

One of the limitations of this research is the data used. I extracted newspaper texts from Delpher, a digital archive of Dutch texts, from the time periods 1945-1954 and 1985-1994. I focused on two time periods of ten years, to be able to collect 100 million words, and on these periods in particular as Dutch spelling remained practically stable and Delpher had the same selection criteria for newspapers between 1945 and 1995. However, forty years is only a short time frame for semantic change. For example, Hamilton et al. (2016b) studied semantic change between 1800 and 1990 and Kulkarni et al. (2015) studies semantic change between 1900 and 2005.

Another limitation regarding the corpus, is that it is not completely random compiled. Delpher order their newspapers per publisher. For example, the first 280 newspapers of 1985 are all newspapers from ’Nieuwe Rotterdamse Courant’, while the first 109 newspapers of 1945 are all from ’Stichting Je Maintiendrai Friesland’. It is thus possible that different newspapers are selected in the different time periods, which can cause skewness. When extracting texts from Delpher, 1000 consecutive newspapers are always extracted. In my code (scraperapiartikel.py), I extracted the first thousand newspapers from Delpher, and then continue with the next thousand, with the code:

```
startrecord = startrecord + 1000
```

It might have been better to extract random numbers that are multiples of a thousand, such as 11000, 8000, 3000, 17000, and to then collect the thousand newspapers starting from those random numbers. This ensures that the corpus is constructed as randomly as possible, which is a requirement for compiling a representative corpus Biber (1993).

A third limitation about the corpus is that it is not entirely part-of-speech tagged, as it took multiple hours to tag a thousand sentences. Due to this, I was unable to automatically extract all verbs from the entire corpus, without having to do it manually.

A final limitation about the corpus is that I examined the quality of the OCR translations on 400 tokens in total. The results of this quality check were quite positive, but maybe not representative of the entire 200 million token corpus. It would have been of added value when a larger subcorpus was examined.

The limitations of the corpus might be responsible for the very high cosine distances. The most changing verb with the highest cosine distance is *volgen* with a cosine distance of 0.938, which implies that the contexts of *volgen* in 1945-1954 and 1985-1994 are as good as completely different. A word that one expects to not have changed is *de*. This word is in all models one of the least changing words, but still has a mean cosine distance of 0.380 in the default model.

Regarding my methodology, most limitations concern topics I have not studied and are therefore interesting for further research. The first limitation is that I have only constructed word embeddings using the skip-gram negative sample architecture. Hamilton et al. (2016b) have also created word embeddings using positive point-wise mutual information (PPMI) and singular value decomposition (SVD). PPMI performs clearly worse than SVD and SGNS, so might not be interesting to study in the context of my study, but SVD performs better on detecting known shifts than SGNS. Other architectures that have been used for word embeddings are skip-gram hierarchical softmax (Haagsma & Nissim, 2017) and continuous bag-of-words (CBOW) (Pierrejean & Tanguy, 2018b).

A second limitation regarding the methodology involves the hyperparameters chosen and studied in the embedding algorithm. I chose to use the hyperparameters of Pierrejean and Tanguy (2018a) for the default model, as the optimal hyperparameter settings depend mostly on corpus size. Pierrejean and Tanguy (2018a) also use corpora containing 100 million words each. To examine the stability of different models, I changed the value of either the window size or the dimension size. Additionally, it is also interesting to vary in the setting of minimal count, depending it on the frequency of tokens in the corpus, and to vary in the setting of negative words, which can be set to 5-20 words for smaller datasets and 2-5 for large datasets (Mikolov, Sutskever, et al., 2013).

More interesting additions to my research relate to the semantic changes and how they can be studied. One of these additions is a more extensive focus on situation types of verbs identified by Vendler (1967). I have briefly mentioned the situation types of the verbs that have changed according to my results, but more can be examined. It would be interesting to tag all verbs with their situation type, especially if this can be done using a computational tool. It this way, it can be studied if different situation types change more or less. A disadvantage of these situation types is that they sometimes depend on the context it is used in. It might therefore be more feasible to look at other types of semantic types, for example those of Cornetto (Maks, Van Der Vliet, Görög, & Vossen, 2013). Cornetto identifies the semantic verb types of state, describing a situation that does not change over time, action, describing an action that is usually controlled by the subject of the verb, and process, describing a dynamic event that is not initiated by an actor capable of acting with volition.

A second addition is the examining of the influence of polysemy on semantic change, as is done by Hamilton et al. (2016b) and discussed by Haagsma and Nissim (2017). Hamilton et al. (2016b) present the law of innovation: polysemous words change at faster rates. It is interesting to test this notion using a Dutch corpus to see if it holds for Dutch. Hamilton measures a word's contextual diversity as a proxy for its degree of polysemy. The degree of polysemy in Dutch words can be measured in such a manner, but can also be extracted from for example Cornetto or Groot woordenboek der Nederlandse taal. Using a dictionary for the number of senses might be more convenient than using Cornetto as dictionaries from different time periods can be consulted, while Cornetto only contains the number of senses at one specific point of time. The verbs in the corpus can be tagged with their degree of polysemy to see which verbs change more of less than other verbs. The problem of compound verbs should be accounted for when tagging word forms, as for example the lemma *maken* has 4 senses, but including the senses of the ten compound verbs, it has 41 senses. It might be better to first lemmatise the corpus, to clarify which verb is actually changing.

Considering lemmatising creates another addition to this research. Some research has challenged the utility of the notion of lemma, arguing that each word form tends to occur in distinctive grammatical contexts, with distinct meanings and uses (Biber, 2012). The standard deviations in Table 28-43 present the variation in cosine distances between word forms corresponding to a particular lemma. Using these standard deviations, information might be deduced on semantic changes in forms versus lemmas, for example if some word forms, such as first person singular present tense, change more or less than other word forms. If this shows that lemmas change, and not forms independently, the entire corpus can first be lemmatised before measuring cosine distances.

A fourth addition to this research is testing how verbs change. Now, it is only examined if verbs have changed or not, and how much, but not how or in which direction. Semantic changes in verbs can then be linked to the classification of twelve types of semantic change set up by Campbell (2013).

A final addition to the methodology of this research is to not only assess the stability of the models, but also the quality, as is studied by Hellrich and Hahn (2016a, 2016b); Levy et al. (2015); Sahlgren and Lenci (2016). The quality of synchronic word embeddings can be tested using word similarity benchmarks (such as WordSim353, MEN, Mechanical Turk, Stanford Rare Words and SimLex-999), analogy tasks ('a is to a* as b is to b*', where b* is hidden and must be guessed), and multiple-choice vocabulary tests (the TOEFL synonyms and the ESL synonyms) (Hellrich & Hahn, 2016a, 2016b; Levy et al., 2015; Sahlgren & Lenci, 2016).

These quality tests all examine synchronic word embeddings, so an extra addition is to see if there is a method for examining the quality of diachronic word embeddings. For that, benchmarks for known changes might to be developed. It might also be possible to use dictionaries of the relevant periods as benchmarks, or look at the concordances of the semantically changed verbs in the corpus to manually assess possible changes in the context of the verb.

In this study, I have measured semantic change by measuring the cosine distance between a word's vectors for two time periods. A second measure that can be used to measure semantic change is the local neighbourhood measure, measuring the extent to which a word's similarity with its nearest neighbours has changed (Hamilton et al., 2016a). According to Hamilton et al. (2016a), this second measure assigns higher rates of semantic change to nouns than to verbs, but as the importance of a word's nearest neighbours occur in many researches, using this measure is a good addition for further research.

Another use of a word's nearest neighbours is presented in Pierrejean and Tanguy (2018a), Chugh et al. (2018) and Hellrich and Hahn (2016a, 2016b). In these papers, the stability of a model is assessed by measuring the variation that exists between several models trained with the same hyperparameters in terms of its nearest neighbours. A word having the same nearest neighbours across several models is thus considered stable.

The variation between these sets of nearest neighbours can be measured with the Jaccard similarity index, which is used in Chugh et al. (2018), Tahmasebi (2018) and Hellrich and Hahn (2016a, 2016b). The Jaccard similarity returns values ranging from zero (no matching) to one (both seth are identical). A value closer to one thus indicates greater consistency in a word's neighbourhood (Chugh et al., 2018). Given two lists of words, the Jaccard similarity measures the overlap between between these lists divided by the number of unique items in both lists.

This use of the Jaccard similarity index is not only interesting for further research on the stability of Dutch word embedding models using the nearest neighbours, but might also be used to compare the top tens of most changing verbs per model. In this way, the Jaccard similarity index can be used to say something about the stability of the models, complementing the stability measured in this study as the measure of stability used now is not based on existed literature. The Jaccard similarity is then applied to the stability of diachronic word embeddings.

In this study, I contribute to previous research by employing a word embedding model on a diachronic Dutch corpus containing 100 million tokens per time period to study semantic changes in Dutch. I also presented a possible method of measuring the stability of semantic changes, distinguishing this study from earlier studies that measured stability in synchronic word embeddings.

References

Aitchison, J. (2013). *Language change: progress or decay?* (4th ed ed.). Oxford ; New York: Cambridge University Press.

Bengio, Y. (2008). Neural net language models. *Scholarpedia*, *3*(1), 3881.

Biber, D. (1993). Representativeness in Corpus Design. *Literary and linguistic computing*, *8*(4), 243–257.

Biber, D. (2012). Corpus-Based and Corpus-driven Analyses of Language Variation and Use. In *The oxford handbook of linguistics analysis* (chap. 8). Oxford University Press.

Bree, C. v. (1996). Betekenisverandering. In *Historische taalkunde* (chap. 8). Leuven/Amersfoort: Acco.

Broekhuis, H., & Corver, N. (2015). *Syntax of Dutch: Verb and Verb Phrases. Volume 2.* Amsterdam University Press.

Campbell, L. (2013). Semantic Change and Lexical Change. In *Historical linguistics : An introduction* (pp. 221–246). Edinburgh: Edinburgh University Press.

Cedergren, H. (1988). The spread of language change: Verifying inferences of linguistic diffusion. *Language spread and language policy: Issues, implications, and case studies*, 45–60.

Chugh, M., Whigham, P. A., & Dick, G. (2018). Stability of Word Embeddings Using Word2Vec. In T. Mitrovic, B. Xue, & X. Li (Eds.), *Ai 2018: Advances in artificial intelligence. ai 2018. lecture notes in computer science, vol 11320.* Springer, Cham.

Clark, S. (2015). Vector Space Models of Lexical Meaning. In *The handbook of contemporary semantic theory* (pp. 493–522).

Delpher. (n.d.). *Wat is Delpher?* Retrieved from `https://www.delpher.nl/nl/platform/pages/helpitems?nid=372`

Del Tredici, M., Nissim, M., & Zaninello, A. (2016). Tracing metaphors in time through self-distance in vector spaces. *CoRR*, *abs/1611.0*.

Firth, J. R., & Palmer, F. R. (1968). *Selected papers of J.R. Firth, 1952-59.* Harlow: Longmans.

Geerts, G., Van Den Broeck, J., & Verdoodt, A. (1977). Successes and Failures in Dutch Spelling Reform. In *Advances in the creation and revision of writing systems* (pp. 179–245). The Hague: Mouton & Co.

Haagsma, H., & Nissim, M. (2017). A Critical Assessment of a Method for Detecting Diachronic Meaning Shifts: Lessons Learnt from Experiments on Dutch. *Computational Linguistics in the Netherlands Journal*, *7*, 65–78.

Hamilton, W. L., Leskovec, J., & Jurafsky, D. (2016a). Cultural Shift or Linguistic Drift? Comparing Two Computational Measures of Semantic Change. In *Proceedings of the conference on empirical methods in natural language processing. conference on empirical methods in natural language processing* (pp. 2116–2121).

Hamilton, W. L., Leskovec, J., & Jurafsky, D. (2016b). *Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change* (Tech. Rep.).

Hellrich, J., & Hahn, U. (2016a). An assessment of experimental protocols for tracing changes in word semantics relative to accuracy and reliability. In *Proceedings of the 10th sighum workshop on language technology for cultural heritage, social sciences, and humanities* (pp. 111–117).

Hellrich, J., & Hahn, U. (2016b). Bad company—neighborhoods in neural embedding spaces considered harmful. In *Proceedings of coling 2016, the 26th international conference on computational linguistics: Technical papers* (pp. 2785–2796).

Hinton, G. (2011). *Lecture 5 : Distributed Representations* (Tech. Rep.).

Hróarsdóttir, T. (2004). Language Change and Language Acquisition. *Nordlyd*, *31*(1).

Jensen, K. (2014). Linguistics and the digital humanities: (computational) corpus linguistics. *MedieKultur: Journal of Media and Communication Research*, *30*(57), 115–134.

Koninklijke Bibliotheek. (n.d.-a). *1945-1995.* Retrieved from `https://www.kb.nl/organisatie/onderzoek-expertise/digitaliseringsprojecten-in-de-kb/project-databank-digitale-dagbladen/geselecteerde-titels-en-selectieprocedure/selectie-van-titels/1945-1995`

Koninklijke Bibliotheek. (n.d.-b). *Selectieproces.* Retrieved from `https://www.kb.nl/organisatie/onderzoek-expertise/digitaliseringsprojecten-in-de-kb/project-databank-digitale-dagbladen/geselecteerde-titels-en-selectieprocedure/selectieproces`

Kulkarni, V., Al-Rfou, R., Perozzi, B., & Skiena, S. (2015). Statistically Significant Detection of Linguistic Change. In *Proceedings of the 24th international conference on world wide web* (pp. 625–635). Florence, Italy: International World Wide Web Conferences Steering

Committee.

Landsbergen, F. (2009). *The syntactic and semantic development of the Dutch verb krijgen* (Tech. Rep.). Retrieved from `https://openaccess.leidenuniv.nl/bitstream/handle/1887/13971/chapter%204.pdf?sequence=9`

Levy, O., Goldberg, Y., & Dagan, I. (2015). Improving Distributional Similarity with Lessons Learned from Word Embeddings. *Transactions of the Association for Computational Linguistics*, *3*, 211–225.

Maks, I., Van Der Vliet, H., Görög, A., & Vossen, P. (2013). *Cornetto LMF Lexical Resource for Dutch* (Tech. Rep.). Retrieved from `http://cornetto.clarin.inl.nl/help/D9-Cornetto-Documentation.pdf`

McCormick, C. (2017). *Word2Vec Tutorial Part 2 - Negative Sampling.* Retrieved from `http://mccormickml.com/2017/01/11/word2vec-tutorial-part-2-negative-sampling/`

Mihalicek, V., & Wilson, C. (Eds.). (2011). *Language files: materials for an introduction to language and linguistics* (11th ed ed.). Columbus: Ohio State University Press.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 3111–3119).

Mikolov, T., Yih, W.-T., & Zweig, G. (2013). Linguistic regularities in continuous space word representations. *Proceedings of NAACL-HLT*(June), 746–751.

Pierrejean, B., & Tanguy, L. (2018a). Predicting Word Embeddings Variability. In *Proceedings of the seventh joint conference on lexical and computational semantics* (pp. 154–159).

Pierrejean, B., & Tanguy, L. (2018b). Towards Qualitative Word Embeddings Evaluation: Measuring Neighbors Variation. In *Conference of the north american chapter of the association for computational linguistics: Student research workshop* (pp. 32–39).

Rehurek, R. (n.d.). *gensim: models.word2vec – Word2vec embeddings.* Retrieved from `https://radimrehurek.com/gensim/models/word2vec.html`

Richards, B. J. (1987). Type/Token Ratios: what do they really tell us? *Journal of Child*

Language, *14*(02), 201–209.

Ruder, S. (2016a). *On word embedding - Part 2: Approximating the Softmax.* Retrieved from http://ruder.io/word-embeddings-softmax/index.html

Ruder, S. (2016b). *On word embeddings - Part 1.* Retrieved from http://ruder.io/word-embeddings-1/

Ruder, S. (2016c). *On word embeddings - Part 3: The secret ingredients of word2vec.* Retrieved from http://ruder.io/secret-word2vec/

Saeed, J. I. (2015). *Semantics.* Wiley-Blackwell.

Sagi, E., Kaufmann, S., & Clark, B. (2011). Tracing semantic change with Latent Semantic Analysis. In K. Allan & J. A. Robinson (Eds.), *Current methods in historical semantics.* Berlin, Boston: De Gruyter.

Sahlgren, M., & Lenci, A. (2016). The Effects of Data Size and Frequency Range on Distributional Semantic Models.

Schroll, R. (n.d.). *NLTK vs. spaCy: Natural Language Processing in Python | The Data Incubator.* Retrieved from https://blog.thedataincubator.com/2016/04/nltk-vs-spacy-natural-language-processing-in-python/

Steyaert, K. (n.d.). *History of Dutch Language.* Retrieved from https://www.ucl.ac.uk/dutchstudies/an/SP_LINKS_UCL_POPUP/SPs_english/language_history/pages/intro.html

Tahmasebi, N. (2018). A Study on Word2Vec on a Historical Swedish Newspaper Corpus. In *Dhn* (pp. 25–37).

Tognini-Bonelli, E. (2001). *Corpus Linguistics at Work* (Vol. 6). Amsterdam: John Benjamins Publishing Company.

Traub, M., Van Ossenbruggen, J., & Hardman, L. (2015). Impact Analysis of OCR Quality on Research Tasks in Digital Archives. In *nternational conference on theory and practice of digital libraries* (pp. 252–263).

Traugott, E. C., & Dasher, R. B. (2004). *Regularity in Semantic Change.* Cambridge: Cambridge University Press.

van Walsum, S. (2017, 3). *Directeur Taalunie: 'Hun hebben' is taalkundig gezien zelfs een verbetering.*

Vendler, Z. (1967). *Linguistics in Philosophy.* Cornell University Press.

Verhagen, A. (1998). *Changes in the use of Dutch doen and the nature of semantic knowledge* (Tech. Rep.). Retrieved from https://openaccess.leidenuniv.nl/bitstream/handle/1887/2387/302_011.pdf?sequence=1

Vossen, P., Maks, I., Segers, R., Van Der Vliet, H., Moens, M.-F., Hofmann, K., ... de Rijke, M. (2013). Cornetto: a combinatorial lexical semantic database for Dutch. In *Essential speech and language technology for dutch* (pp. 165–184). Springer.

Appendix A

Stop words

a, aan, al, alles, als, altijd, andere, b, ben, bij, c, d, daar, dan, dat, de, den, der, deze, die, dit, doch, doen, door, dus, e, een, eens, en, er, f, g, ge, geen, geweest, h, haar, had, heb, hebben, heeft, hem, het, hier, hij, hoe, hun, i, iemand, iets, ik, in, is, j, ja, je, jij, k, kan, kon, kunnen, l, m, maar, me, meer, men, met, mij, mijn, moet, n, na, naar, niet, niets, nog, nu, o, of, om, omdat, onder, ons, ook, op, over, p, q, r, reeds, s, t, te, tegen, toch, toen, tot, u, uit, uw, v, van, veel, voor, w, want, waren, was, wat, we, werd, wezen, wie, wij, wil, worden, wordt, x, y, z, zal, ze, zelf, zich, zij, zijn, zo, zonder, zou

Appendix B

The most changing verbs and the amount of runs they changed in per model

Table B1

The most changing verbs and the amount of runs they changed in for the default model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
gegroeid	4	*lachen*	2	*kijkt*	1
groeide	4	*nagelaten*	2	*leen*	1
onderscheiden	4	*ontbreekt*	2	*lijdt*	1
volgen	4	*bleken*	1	*ontmoeten*	1
behandelt	3	*bood*	1	*raden*	1
keert	3	*brak*	1	*schrijft*	1
luister	3	*draaide*	1	*ontstaan*	1
bekeken	2	*eindigde*	1	*vroegen*	1
inzien	2	*geboden*	1	*wint*	1

Table B2

The most changing verbs and the amount of runs they changed in for the WIN1 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
wilden	5	*verkeert*	2	*ontsnapt*	1
zond	5	*verschijnt*	2	*speelt*	1
hef	4	*beoogt*	1	*opgeven*	1
opgewonden	4	*binnengekomen*	1	*rijdt*	1
antwoordde	3	*dalen*	1	*schaken*	1
gewend	3	*int*	1	*schreven*	1
luidt	3	*kiest*	1	*vertoont*	1
ophouden	3	*levert*	1	*wint*	1
omvat	2				

Table B3

The most changing verbs and the amount of runs they changed in for the WIN2 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
besteld	5	*draagt*	2	*leidt*	1
gerechtigd	5	*wint*	2	*luister*	1
kiest	5	*woont*	2	*schieten*	1
omvat	4	*aangegeven*	1	*toonde*	1
uitgeven	4	*belooft*	1	*breken*	1
bedacht	3	*binnengekomen*	1	*uitkomen*	1
koken	3	*gegooid*	1	*verwijderd*	1
zond	3	*gepasseerd*	1		

Table B4

The most changing verbs and the amount of runs they changed in for the WIN3 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
bewaren	3	*trouwen*	2	*meldde*	1
luister	3	*vroegen*	2	*ondergaan*	1
volgt	3	*probeert*	2	*ontwikkelen*	1
dien	2	*opgeheven*	2	*ophouden*	1
eet	2	*opgegeven*	1	*schieten*	1
eist	2	*brengt*	1	*speelden*	1
gelopen	2	*droeg*	1	*springen*	1
geschoten	2	*gebroken*	1	*studeerde*	1
ingegaan	2	*gold*	1	*verkeert*	1
legt	2	*groeit*	1	*vervolgt*	1
opnemen	2	*ingenomen*	1		

Table B5

The most changing verbs and the amount of runs they changed in for the WIN4 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
uitbrak	3	*verkeert*	1	*meldde*	1
rusten	3	*verdwijnen*	1	*leen*	1
dreigde	3	*verdragen*	1	*gewezen*	1
zendt	2	*stoppen*	1	*geslaagd*	1
neemt	2	*stelden*	1	*geleid*	1
groeide	2	*schiet*	1	*droeg*	1
behandelt	2	*richt*	1	*draagt*	1
bedacht	2	*opnemen*	1	*denkt*	1
aanvaarden	2	*ophouden*	1	*bood*	1
zwemmen	1	*opgewonden*	1	*bleven*	1
zingen	1	*opgeleverd*	1	*bevrijd*	1
volgt	1	*ontwikkeld*	1	*bestaan*	1
verschenen	1	*ontbrak*	1		

Table B6

The most changing verbs and the amount of runs they changed in for the WIN6 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
ontworpen	5	*heette*	2	*neerleggen*	1
opgeheven	4	*aangegeven*	1	*opgegeven*	1
treffen	4	*bedacht*	1	*schaken*	1
behandeld	3	*beleefd*	1	*verdwijnen*	1
inzien	3	*bestrijden*	1	*verzorgen*	1
leert	3	*betrekken*	1	*voert*	1
uitbrak	3	*bleken*	1	*volgt*	1
afgeleverd	2	*heten*	1	*voorzien*	1
bezorgd	2	*luister*	1	*zendt*	1
bood	2				

Table B7

The most changing verbs and the amount of runs they changed in for the WIN7 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
behandeld	5	*getrokken*	2	*geschieden*	1
onderscheiden	4	*onderzocht*	2	*gevormd*	1
volgen	4	*overgegaan*	2	*haalde*	1
begrepen	3	*wint*	2	*juist*	1
luister	3	*betekenen*	1	*lachen*	1
toont	3	*bezien*	1	*ontmoet*	1
brak	2	*draaien*	1	*opgeleverd*	1
draait	2	*eist*	1	*richt*	1
genoten	2	*gekregen*	1	*verbieden*	1

Table B8

The most changing verbs and the amount of runs they changed in for the WIN8 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
doorgaan	4	*gelopen*	2	*ingeschakeld*	1
drijven	4	*geschonden*	2	*luidde*	1
eist	4	*ontsnapt*	2	*ophouden*	1
bekeken	3	*bedacht*	1	*plaatst*	1
belegd	3	*bleven*	1	*schaken*	1
gevuld	3	*gebonden*	1	*verkeert*	1
luister	3	*gesproken*	1	*verminderen*	1
opgeroepen	3	*gold*	1	*vertoonde*	1
voorkomt	3	*hielden*	1		

Table B9

The most changing verbs and the amount of runs they changed in for the WIN9 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
lachen	4	*verbieden*	2	*steekt*	1
leert	4	*besteden*	1	*uitoefenen*	1
sluiten	4	*gelopen*	1	*verdwijnen*	1
bedacht	3	*geschonden*	1	*verleent*	1
bleven	3	*getrokken*	1	*verrichten*	1
droeg	3	*gewerkt*	1	*verschijnen*	1
rusten	3	*inzien*	1	*voorkomt*	1
eist	2	*kiest*	1	*voorspellen*	1
heelt	2	*maak*	1	*waarschuwen*	1
leen	2	*scoorde*	1		

Table B10

The most changing verbs and the amount of runs they changed in for the WIN10 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
lijdt	4	*verkeert*	2	*luister*	1
doorgaan	3	*veroorzaken*	2	*noemt*	1
vervolgt	3	*bedacht*	1	*rusten*	1
volgt	3	*draait*	1	*schrijft*	1
bekeken	2	*eindigt*	1	*sparen*	1
brak	2	*eist*	1	*verklaart*	1
gelaten	2	*gebracht*	1	*verwerken*	1
gewenst	2	*gehandeld*	1	*wekken*	1
keert	2	*gelezen*	1	*wint*	1
rijdt	2	*gelopen*	1	*zingen*	1
roepen	2	*lachen*	1		

Table B11

The most changing verbs and the amount of runs they changed in for the DIM50 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
volgen	5	*gebonden*	2	*bood*	1
eist	4	*kiest*	2	*dalen*	1
gelopen	4	*ontbreken*	2	*geregeld*	1
toont	4	*slaat*	2	*gestaan*	1
uitkomen	4	*vertoont*	2	*meldde*	1
afgebroken	3	*beslist*	1	*voorgeschreven*	1
onderscheiden	3	*bleken*	1	*zochten*	1
verklaart	3	*bleven*	1		

Table B12

The most changing verbs and the amount of runs they changed in for the DIM200 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
loopt	5	*vertoonde*	2	*luidde*	1
droegen	4	*bedoeld*	1	*ondergaan*	1
opnemen	4	*bepalen*	1	*ontbreekt*	1
slaagde	4	*draait*	1	*ontworpen*	1
verenigen	4	*gebleken*	1	*springen*	1
gebonden	3	*gedwongen*	1	*vasthouden*	1
gekregen	3	*geprobeerd*	1	*verstaan*	1
rusten	2	*kloppen*	1	*volgen*	1
schieten	2	*kocht*	1	*wint*	1

Table B13

The most changing verbs and the amount of runs they changed in for the DIM300 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
verwijderd	3	*droeg*	1	*ontmoeten*	1
omschreven	2	*eist*	1	*opgeroepen*	1
opgebouwd	2	*geuit*	1	*rusten*	1
probeert	2	*heelt*	1	*slaagde*	1
verliest	2	*horen*	1	*spraken*	1
voorgeschreven	2	*ingeschakeld*	1	*stelt*	1
vroegen	2	*keert*	1	*stoppen*	1
zochten	2	*kijkt*	1	*verdwenen*	1
aangetoond	1	*krijg*	1	*verlopen*	1
bedoeld	1	*leidde*	1	*veroverde*	1
behandeld	1	*luister*	1	*vond*	1
behoeven	1	*meldde*	1	*vullen*	1
bewaard	1	*mogen*	1	*wenste*	1
bleken	1	*neem*	1		

Table B14

The most changing verbs and the amount of runs they changed in for the DIM400 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
luister	5	*rusten*	2	*meegenomen*	1
verliest	4	*vluchten*	2	*opgeroepen*	1
afgebroken	3	*bedacht*	1	*rijdt*	1
afgewezen	3	*bezorgd*	1	*trekt*	1
geleverd	3	*bleken*	1	*uitgeroepen*	1
vertrouwde	3	*droegen*	1	*verenigen*	1
aanvaarden	2	*erkent*	1	*verschijnen*	1
bewijst	2	*gelopen*	1	*verwijderd*	1
gevolgd	2	*geniet*	1	*werkt*	1
opgegeven	2	*durft*	1		

Table B15

The most changing verbs and the amount of runs they changed in for the DIM500 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
kijkt	5	*vertrouwde*	2	*kocht*	1
aangetoond	4	*zond*	2	*krijg*	1
geleverd	3	*bekeken*	1	*overgedragen*	1
heelt	3	*betekenen*	1	*raakt*	1
herhaalde	3	*droeg*	1	*schiet*	1
aanvaarden	2	*eet*	1	*slaat*	1
afgebroken	2	*gekregen*	1	*trouwen*	1
eist	2	*gevochten*	1	*verkregen*	1
overwegen	2	*gezocht*	1	*wenste*	1
verliest	2	*ingaan*	1	*werkt*	1

Table B16

The most changing verbs and the amount of runs they changed in for the DIM600 model

Word	Amount of runs	Word	Amount of runs	Word	Amount of runs
gelopen	5	*binnengekomen*	1	*klinkt*	1
luister	4	*denk*	1	*leidde*	1
beantwoorden	2	*droegen*	1	*meegenomen*	1
koos	2	*eindigde*	1	*meende*	1
staken	2	*gedwongen*	1	*ophouden*	1
trekt	2	*gehandeld*	1	*overwegen*	1
verleent	2	*gehoord*	1	*starten*	1
verricht	2	*geprobeerd*	1	*uitkomen*	1
verwerven	2	*hoorde*	1	*verliest*	1
wenste	2	*ingrijpen*	1	*voorgeschreven*	1
afgebroken	1	*gekend*	1	*zendt*	1
betaalde	1	*kijkt*	1		

Appendix C

The cosine distances of the verb forms per lemma per model [47]

Table C1

The cosine distances of all verb forms of zijn, worden and hebben for the default model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.564	0.035	*geworden*	0.614	0.031	*gehad*	0.678	0.041
bent	0.537	0.034	*werd*	0.442	0.019	*had*	0.542	0.026
geweest	0.693	0.019	*werden*	0.532	0.024	*hadden*	0.501	0.043
is	0.379	0.029	*word*	0.640	0.008	*heb*	0.462	0.031
waren	0.640	0.031	*worden*	0.405	0.019	*hebben*	0.430	0.014
was	0.498	0.022	*wordt*	0.499	0.021	*hebt*	0.591	0.026
zijn	0.477	0.066				*heeft*	0.557	0.015

[47] Some verb forms do not occur in this table, because they appear less than 100 times in the corpus of one or both time period(s). These forms are: *breek* and *beweeg*.

Table C2

The cosine distances of all verb forms of komen, gaan and maken for the default model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.600	0.020	*ga*	0.613	0.015	*gemaakt*	0.679	0.030
kom	0.682	0.024	*gaan*	0.536	0.010	*maak*	0.706	0.045
komen	0.531	0.025	*gaat*	0.497	0.034	*maakt*	0.693	0.028
komt	0.481	0.024	*gegaan*	0.665	0.048	*maakte*	0.716	0.030
kwam	0.581	0.023	*ging*	0.602	0.050	*maakten*	0.617	0.023
kwamen	0.551	0.018	*gingen*	0.583	0.033	*maken*	0.725	0.021

Table C3

The cosine distances of all verb forms of geven, doen and houden for the default model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.633	0.042	*deden*	0.669	0.029	*gehouden*	0.676	0.032
gaven	0.737	0.019	*deed*	0.689	0.011	*hield*	0.641	0.040
geef	0.614	0.014	*doe*	0.662	0.027	*hielden*	0.797	0.030
geeft	0.686	0.043	*doen*	0.598	0.022	*hou*	0.745	0.036
gegeven	0.670	0.023	*doet*	0.620	0.049	*houd*	0.759	0.061
geven	0.594	0.014	*gedaan*	0.612	0.028	*houden*	0.615	0.023
						houdt	0.558	0.009

Table C4

The cosine distances of all verb forms of zeggen, zien and krijgen for the default model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.618	0.036	*gezien*	0.785	0.031	*gekregen*	0.571	0.033
zeg	0.724	0.028	*zag*	0.522	0.024	*kreeg*	0.595	0.026
zeggen	0.622	0.023	*zagen*	0.804	0.029	*kregen*	0.692	0.021
zegt	0.626	0.030	*zie*	0.784	0.024	*krijg*	0.854	0.012
zei	0.606	0.013	*zien*	0.616	0.012	*krijgen*	0.686	0.017
zeiden	0.634	0.061	*ziet*	0.692	0.060	*krijgt*	0.770	0.089

Table C5

The cosine distances of all verb forms of bieden, breken and bewegen for the default model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.754	0.036	*brak*	0.850	0.074	*beweegt*	0.848	0.063
bieden	0.606	0.034	*braken*	0.722	0.034	*bewegen*	0.676	0.021
biedt	0.679	0.041	*breekt*	0.838	0.021	*bewogen*	0.716	0.057
boden	0.772	0.034	*breken*	0.637	0.032	*bewoog*	0.818	0.008
bood	0.870	0.021	*gebroken*	0.733	0.050			
geboden	0.848	0.037						

Table C6

The cosine distances of all verb forms of zijn, worden and hebben for the WIN1 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.688	0.039	*geworden*	0.599	0.036	*gehad*	0.552	0.050
bent	0.687	0.013	*werd*	0.597	0.018	*had*	0.640	0.026
geweest	0.629	0.055	*werden*	0.484	0.042	*hadden*	0.593	0.052
is	0.582	0.017	*word*	0.697	0.056	*heb*	0.595	0.046
waren	0.590	0.055	*worden*	0.531	0.018	*hebben*	0.506	0.059
was	0.645	0.052	*wordt*	0.624	0.042	*hebt*	0.751	0.052
zijn	0.696	0.037				*heeft*	0.594	0.029

Table C7

The cosine distances of all verb forms of komen, gaan and maken for the WIN1 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.599	0.025	*ga*	0.675	0.022	*gemaakt*	0.565	0.049
kom	0.641	0.035	*gaan*	0.553	0.019	*maak*	0.545	0.033
komen	0.626	0.023	*gaat*	0.563	0.046	*maakt*	0.732	0.041
komt	0.698	0.022	*gegaan*	0.620	0.032	*maakte*	0.733	0.008
kwam	0.583	0.025	*ging*	0.587	0.015	*maakten*	0.575	0.031
kwamen	0.552	0.021	*gingen*	0.692	0.027	*maken*	0.483	0.021

Table C8

The cosine distances of all verb forms of geven, doen and houden for the WIN1 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.659	0.014	*deden*	0.698	0.034	*gehouden*	0.489	0.057
gaven	0.653	0.043	*deed*	0.556	0.027	*hield*	0.630	0.021
geef	0.662	0.050	*doe*	0.565	0.021	*hielden*	0.702	0.017
geeft	0.681	0.041	*doen*	0.556	0.029	*hou*	0.699	0.015
gegeven	0.710	0.027	*doet*	0.601	0.041	*houd*	0.662	0.018
geven	0.447	0.026	*gedaan*	0.586	0.019	*houden*	0.525	0.044
						houdt	0.609	0.021

Table C9

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN1 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.530	0.010	*gezien*	0.641	0.045	*gekregen*	0.701	0.033
zeg	0.625	0.025	*zag*	0.629	0.024	*kreeg*	0.607	0.028
zeggen	0.468	0.041	*zagen*	0.737	0.037	*kregen*	0.663	0.016
zegt	0.538	0.047	*zie*	0.654	0.041	*krijg*	0.513	0.025
zei	0.644	0.052	*zien*	0.546	0.101	*krijgen*	0.631	0.020
zeiden	0.566	0.008	*ziet*	0.619	0.045	*krijgt*	0.571	0.020

Table C10

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN1 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.654	0.020	*brak*	0.813	0.051	*beweegt*	0.642	0.028
bieden	0.663	0.012	*braken*	0.490	0.008	*bewegen*	0.631	0.031
biedt	0.711	0.033	*breekt*	0.716	0.052	*bewogen*	0.694	0.017
boden	0.707	0.033	*breken*	0.711	0.021	*bewoog*	0.641	0.051
bood	0.744	0.057	*gebroken*	0.734	0.027			
geboden	0.670	0.047						

Table C11

The cosine distances of all verb forms of zijn, worden and hebben for the WIN2 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.572	0.031	*geworden*	0.628	0.018	*gehad*	0.563	0.026
bent	0.652	0.039	*werd*	0.570	0.038	*had*	0.496	0.035
geweest	0.600	0.035	*werden*	0.587	0.063	*hadden*	0.558	0.040
is	0.464	0.020	*word*	0.615	0.010	*heb*	0.557	0.026
waren	0.538	0.069	*worden*	0.582	0.020	*hebben*	0.569	0.045
was	0.561	0.047	*wordt*	0.642	0.041	*hebt*	0.614	0.022
zijn	0.568	0.032				*heeft*	0.620	0.036

Table C12

The cosine distances of all verb forms of komen, gaan and maken for the WIN2 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.630	0.031	*ga*	0.573	0.020	*gemaakt*	0.602	0.012
kom	0.648	0.025	*gaan*	0.558	0.010	*maak*	0.631	0.040
komen	0.503	0.027	*gaat*	0.616	0.014	*maakt*	0.698	0.022
komt	0.601	0.032	*gegaan*	0.703	0.033	*maakte*	0.590	0.033
kwam	0.593	0.038	*ging*	0.550	0.019	*maakten*	0.640	0.028
kwamen	0.593	0.020	*gingen*	0.577	0.027	*maken*	0.469	0.045

Table C13

The cosine distances of all verb forms of geven, doen and houden for the WIN2 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.592	0.041	*deden*	0.725	0.044	*gehouden*	0.576	0.044
gaven	0.650	0.048	*deed*	0.710	0.052	*hield*	0.733	0.011
geef	0.791	0.081	*doe*	0.501	0.028	*hielden*	0.792	0.030
geeft	0.638	0.051	*doen*	0.565	0.031	*hou*	0.691	0.025
gegeven	0.481	0.028	*doet*	0.614	0.026	*houd*	0.695	0.025
geven	0.459	0.040	*gedaan*	0.717	0.021	*houden*	0.540	0.047
						houdt	0.646	0.030

Table C14

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN2 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.558	0.043	*gezien*	0.684	0.032	*gekregen*	0.600	0.032
zeg	0.656	0.047	*zag*	0.487	0.037	*kreeg*	0.500	0.045
zeggen	0.487	0.031	*zagen*	0.697	0.025	*kregen*	0.657	0.059
zegt	0.633	0.016	*zie*	0.679	0.032	*krijg*	0.681	0.038
zei	0.652	0.026	*zien*	0.562	0.040	*krijgen*	0.613	0.013
zeiden	0.690	0.036	*ziet*	0.622	0.049	*krijgt*	0.718	0.012

Table C15

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN2 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.665	0.021	*brak*	0.757	0.032	*beweegt*	0.671	0.027
bieden	0.582	0.044	*braken*	0.703	0.018	*bewegen*	0.629	0.028
biedt	0.677	0.045	*breekt*	0.848	0.022	*bewogen*	0.695	0.029
boden	0.759	0.022	*breken*	0.784	0.069	*bewoog*	0.690	0.031
bood	0.713	0.029	*gebroken*	0.793	0.018			
geboden	0.755	0.021						

Table C16

The cosine distances of all verb forms of zijn, worden and hebben for the WIN3 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.615	0.012	*geworden*	0.572	0.066	*gehad*	0.669	0.029
bent	0.602	0.010	*werd*	0.367	0.062	*had*	0.494	0.035
geweest	0.604	0.040	*werden*	0.572	0.014	*hadden*	0.508	0.042
is	0.498	0.017	*word*	0.627	0.049	*heb*	0.512	0.078
waren	0.468	0.064	*worden*	0.414	0.044	*hebben*	0.515	0.056
was	0.495	0.047	*wordt*	0.567	0.048	*hebt*	0.645	0.019
zijn	0.569	0.016				*heeft*	0.546	0.015

Table C17

The cosine distances of all verb forms of komen, gaan and maken for the WIN3 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.632	0.032	*ga*	0.522	0.022	*gemaakt*	0.487	0.034
kom	0.686	0.089	*gaan*	0.537	0.026	*maak*	0.724	0.057
komen	0.564	0.087	*gaat*	0.631	0.076	*maakt*	0.630	0.026
komt	0.602	0.036	*gegaan*	0.608	0.064	*maakte*	0.537	0.042
kwam	0.506	0.036	*ging*	0.536	0.048	*maakten*	0.672	0.029
kwamen	0.542	0.019	*gingen*	0.629	0.040	*maken*	0.538	0.038

Table C18

The cosine distances of all verb forms of geven, doen and houden for the WIN3 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.578	0.051	*deden*	0.691	0.052	*gehouden*	0.633	0.029
gaven	0.667	0.083	*deed*	0.567	0.049	*hield*	0.700	0.076
geef	0.559	0.045	*doe*	0.535	0.012	*hielden*	0.769	0.039
geeft	0.776	0.052	*doen*	0.634	0.041	*hou*	0.713	0.024
gegeven	0.615	0.029	*doet*	0.601	0.030	*houd*	0.692	0.057
geven	0.601	0.048	*gedaan*	0.583	0.021	*houden*	0.603	0.055
						houdt	0.630	0.021

Table C19

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN3 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.569	0.027	*gezien*	0.580	0.047	*gekregen*	0.639	0.053
zeg	0.655	0.036	*zag*	0.623	0.040	*kreeg*	0.633	0.028
zeggen	0.495	0.026	*zagen*	0.637	0.030	*kregen*	0.712	0.067
zegt	0.705	0.037	*zie*	0.803	0.033	*krijg*	0.734	0.018
zei	0.636	0.029	*zien*	0.606	0.045	*krijgen*	0.632	0.046
zeiden	0.735	0.044	*ziet*	0.605	0.054	*krijgt*	0.746	0.029

Table C20

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN3 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.606	0.021	*brak*	0.702	0.021	*beweegt*	0.785	0.044
bieden	0.600	0.027	*braken*	0.689	0.013	*bewegen*	0.663	0.087
biedt	0.598	0.025	*breekt*	0.803	0.044	*bewogen*	0.710	0.068
boden	0.912	0.055	*breken*	0.693	0.043	*bewoog*	0.763	0.035
bood	0.767	0.041	*gebroken*	0.814	0.071			
geboden	0.765	0.048						

Table C21

The cosine distances of all verb forms of zijn, worden and hebben for the WIN4 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.614	0.023	*geworden*	0.682	0.040	*gehad*	0.625	0.027
bent	0.629	0.036	*werd*	0.496	0.047	*had*	0.589	0.042
geweest	0.691	0.022	*werden*	0.579	0.027	*hadden*	0.561	0.043
is	0.443	0.017	*word*	0.619	0.071	*heb*	0.484	0.007
waren	0.533	0.026	*worden*	0.491	0.047	*hebben*	0.539	0.048
was	0.531	0.042	*wordt*	0.562	0.028	*hebt*	0.667	0.045
zijn	0.542	0.013				*heeft*	0.450	0.058

Table C22

The cosine distances of all verb forms of komen, gaan and maken for the WIN4 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.691	0.031	*ga*	0.497	0.026	*gemaakt*	0.634	0.036
kom	0.593	0.024	*gaan*	0.547	0.010	*maak*	0.612	0.056
komen	0.567	0.031	*gaat*	0.583	0.050	*maakt*	0.619	0.021
komt	0.562	0.029	*gegaan*	0.680	0.056	*maakte*	0.620	0.023
kwam	0.529	0.015	*ging*	0.635	0.045	*maakten*	0.677	0.039
kwamen	0.598	0.050	*gingen*	0.704	0.076	*maken*	0.574	0.025

Table C23

The cosine distances of all verb forms of geven, doen and houden for the WIN4 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.591	0.068	*deden*	0.768	0.019	*gehouden*	0.583	0.020
gaven	0.641	0.018	*deed*	0.643	0.047	*hield*	0.708	0.027
geef	0.573	0.025	*doe*	0.524	0.008	*hielden*	0.701	0.051
geeft	0.688	0.053	*doen*	0.588	0.053	*hou*	0.710	0.033
gegeven	0.719	0.026	*doet*	0.612	0.030	*houd*	0.751	0.050
geven	0.537	0.042	*gedaan*	0.632	0.057	*houden*	0.605	0.027
						houdt	0.607	0.030

Table C24

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN4 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.619	0.035	*gezien*	0.752	0.051	*gekregen*	0.774	0.020
zeg	0.706	0.016	*zag*	0.516	0.028	*kreeg*	0.573	0.045
zeggen	0.562	0.048	*zagen*	0.797	0.045	*kregen*	0.749	0.070
zegt	0.725	0.027	*zie*	0.697	0.029	*krijg*	0.724	0.048
zei	0.594	0.059	*zien*	0.522	0.059	*krijgen*	0.751	0.053
zeiden	0.616	0.021	*ziet*	0.672	0.084	*krijgt*	0.709	0.029

Table C25

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN4 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.647	0.035	*brak*	0.776	0.029	*beweegt*	0.846	0.056
bieden	0.674	0.051	*braken*	0.737	0.042	*bewegen*	0.660	0.023
biedt	0.699	0.071	*breekt*	0.880	0.080	*bewogen*	0.753	0.036
boden	0.770	0.039	*breken*	0.695	0.056	*bewoog*	0.729	0.006
bood	0.849	0.022	*gebroken*	0.744	0.024			
geboden	0.815	0.057						

Table C26

The cosine distances of all verb forms of zijn, worden and hebben for the WIN6 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.546	0.023	*geworden*	0.613	0.029	*gehad*	0.599	0.045
bent	0.637	0.023	*werd*	0.414	0.017	*had*	0.510	0.026
geweest	0.649	0.025	*werden*	0.472	0.026	*hadden*	0.603	0.008
is	0.307	0.025	*word*	0.678	0.053	*heb*	0.484	0.006
waren	0.528	0.014	*worden*	0.447	0.024	*hebben*	0.397	0.013
was	0.451	0.015	*wordt*	0.519	0.040	*hebt*	0.570	0.026
zijn	0.467	0.021				*heeft*	0.435	0.019

Table C27

The cosine distances of all verb forms of komen, gaan and maken for the WIN6 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.623	0.026	*ga*	0.579	0.030	*gemaakt*	0.600	0.042
kom	0.807	0.048	*gaan*	0.592	0.029	*maak*	0.661	0.055
komen	0.588	0.033	*gaat*	0.582	0.021	*maakt*	0.668	0.019
komt	0.550	0.027	*gegaan*	0.698	0.038	*maakte*	0.744	0.043
kwam	0.622	0.030	*ging*	0.598	0.054	*maakten*	0.537	0.038
kwamen	0.566	0.039	*gingen*	0.628	0.072	*maken*	0.467	0.043

Table C28

The cosine distances of all verb forms of geven, doen and houden for the WIN6 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.680	0.019	*deden*	0.636	0.021	*gehouden*	0.642	0.040
gaven	0.688	0.023	*deed*	0.794	0.013	*hield*	0.703	0.033
geef	0.570	0.026	*doe*	0.537	0.021	*hielden*	0.769	0.015
geeft	0.635	0.038	*doen*	0.651	0.037	*hou*	0.641	0.021
gegeven	0.797	0.022	*doet*	0.673	0.052	*houd*	0.682	0.037
geven	0.498	0.011	*gedaan*	0.598	0.025	*houden*	0.663	0.041
						houdt	0.673	0.038

Table C29

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN6 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.594	0.035	*gezien*	0.790	0.023	*gekregen*	0.746	0.023
zeg	0.621	0.026	*zag*	0.653	0.041	*kreeg*	0.691	0.017
zeggen	0.587	0.016	*zagen*	0.854	0.022	*kregen*	0.598	0.040
zegt	0.597	0.020	*zie*	0.710	0.039	*krijg*	0.695	0.032
zei	0.576	0.045	*zien*	0.564	0.020	*krijgen*	0.655	0.027
zeiden	0.598	0.080	*ziet*	0.630	0.011	*krijgt*	0.701	0.065

Table C30

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN6 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.784	0.010	*brak*	0.722	0.062	*beweegt*	0.777	0.058
bieden	0.561	0.031	*braken*	0.750	0.017	*bewegen*	0.735	0.023
biedt	0.576	0.036	*breekt*	0.764	0.039	*bewogen*	0.691	0.049
boden	0.724	0.021	*breken*	0.695	0.039	*bewoog*	0.694	0.022
bood	0.904	0.021	*gebroken*	0.664	0.025			
geboden	0.721	0.048						

Table C31

The cosine distances of all verb forms of zijn, worden and hebben for the WIN7 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.478	0.024	*geworden*	0.659	0.024	*gehad*	0.693	0.085
bent	0.604	0.018	*werd*	0.455	0.027	*had*	0.642	0.024
geweest	0.630	0.039	*werden*	0.514	0.043	*hadden*	0.584	0.029
is	0.378	0.028	*word*	0.689	0.042	*heb*	0.519	0.035
waren	0.596	0.033	*worden*	0.417	0.020	*hebben*	0.411	0.015
was	0.376	0.013	*wordt*	0.445	0.031	*hebt*	0.685	0.041
zijn	0.410	0.021				*heeft*	0.456	0.009

Table C32

The cosine distances of all verb forms of komen, gaan and maken for the WIN7 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.679	0.012	*ga*	0.584	0.031	*gemaakt*	0.597	0.017
kom	0.705	0.016	*gaan*	0.693	0.016	*maak*	0.713	0.036
komen	0.488	0.035	*gaat*	0.562	0.028	*maakt*	0.627	0.036
komt	0.570	0.021	*gegaan*	0.687	0.069	*maakte*	0.795	0.050
kwam	0.594	0.048	*ging*	0.603	0.018	*maakten*	0.676	0.028
kwamen	0.636	0.034	*gingen*	0.763	0.009	*maken*	0.587	0.023

Table C33

The cosine distances of all verb forms of geven, doen and houden for the WIN7 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.671	0.019	*deden*	0.754	0.052	*gehouden*	0.557	0.047
gaven	0.647	0.026	*deed*	0.750	0.013	*hield*	0.524	0.018
geef	0.522	0.018	*doe*	0.533	0.058	*hielden*	0.840	0.018
geeft	0.624	0.029	*doen*	0.671	0.021	*hou*	0.642	0.025
gegeven	0.768	0.035	*doet*	0.687	0.068	*houd*	0.667	0.020
geven	0.624	0.034	*gedaan*	0.598	0.026	*houden*	0.606	0.024
						houdt	0.637	0.019

Table C34

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN7 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.587	0.024	*gezien*	0.835	0.043	*gekregen*	0.822	0.063
zeg	0.612	0.055	*zag*	0.662	0.010	*kreeg*	0.664	0.040
zeggen	0.687	0.047	*zagen*	0.787	0.060	*kregen*	0.681	0.015
zegt	0.562	0.031	*zie*	0.612	0.068	*krijg*	0.789	0.010
zei	0.687	0.011	*zien*	0.769	0.043	*krijgen*	0.659	0.042
zeiden	0.564	0.069	*ziet*	0.727	0.028	*krijgt*	0.792	0.057

Table C35

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN7 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.837	0.050	*brak*	0.941	0.015	*beweegt*	0.865	0.050
bieden	0.671	0.022	*braken*	0.811	0.039	*bewegen*	0.758	0.048
biedt	0.679	0.020	*breekt*	0.815	0.047	*bewogen*	0.772	0.022
boden	0.739	0.065	*breken*	0.613	0.057	*bewoog*	0.745	0.059
bood	0.866	0.027	*gebroken*	0.793	0.036			
geboden	0.660	0.016						

Table C36

The cosine distances of all verb forms of zijn, worden and hebben for the WIN8 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.614	0.023	*geworden*	0.642	0.034	*gehad*	0.710	0.022
bent	0.580	0.025	*werd*	0.460	0.018	*had*	0.543	0.029
geweest	0.580	0.019	*werden*	0.552	0.012	*hadden*	0.654	0.017
is	0.393	0.017	*word*	0.622	0.027	*heb*	0.440	0.033
waren	0.595	0.008	*worden*	0.441	0.020	*hebben*	0.443	0.037
was	0.507	0.038	*wordt*	0.459	0.010	*hebt*	0.680	0.029
zijn	0.385	0.025				*heeft*	0.464	0.029

Table C37

The cosine distances of all verb forms of komen, gaan and maken for the WIN8 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.755	0.023	*ga*	0.523	0.017	*gemaakt*	0.566	0.040
kom	0.694	0.021	*gaan*	0.573	0.020	*maak*	0.705	0.018
komen	0.589	0.024	*gaat*	0.611	0.046	*maakt*	0.674	0.031
komt	0.575	0.012	*gegaan*	0.675	0.036	*maakte*	0.755	0.035
kwam	0.465	0.017	*ging*	0.530	0.009	*maakten*	0.713	0.026
kwamen	0.561	0.027	*gingen*	0.757	0.036	*maken*	0.621	0.031

Table C38

The cosine distances of all verb forms of geven, doen and houden for the WIN8 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.650	0.034	*deden*	0.771	0.036	*gehouden*	0.648	0.037
gaven	0.685	0.033	*deed*	0.601	0.026	*hield*	0.569	0.031
geef	0.636	0.012	*doe*	0.539	0.015	*hielden*	0.859	0.095
geeft	0.680	0.028	*doen*	0.609	0.021	*hou*	0.680	0.038
gegeven	0.768	0.044	*doet*	0.620	0.031	*houd*	0.706	0.020
geven	0.680	0.048	*gedaan*	0.619	0.017	*houden*	0.702	0.016
						houdt	0.714	0.027

Table C39

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN8 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.554	0.021	*gezien*	0.679	0.027	*gekregen*	0.665	0.061
zeg	0.664	0.046	*zag*	0.688	0.005	*kreeg*	0.700	0.021
zeggen	0.544	0.030	*zagen*	0.903	0.028	*kregen*	0.580	0.014
zegt	0.561	0.024	*zie*	0.600	0.016	*krijg*	0.729	0.037
zei	0.692	0.052	*zien*	0.598	0.044	*krijgen*	0.705	0.071
zeiden	0.525	0.029	*ziet*	0.598	0.010	*krijgt*	0.800	0.034

Table C40

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN8 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.762	0.015	*brak*	0.755	0.012	*beweegt*	0.736	0.029
bieden	0.658	0.017	*braken*	0.925	0.029	*bewegen*	0.759	0.025
biedt	0.573	0.038	*breekt*	0.878	0.037	*bewogen*	0.711	0.027
boden	0.909	0.042	*breken*	0.600	0.020	*bewoog*	0.682	0.048
bood	0.827	0.042	*gebroken*	0.702	0.020			
geboden	0.915	0.041						

Table C41

The cosine distances of all verb forms of zijn, worden and hebben for the WIN9 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.586	0.015	*geworden*	0.686	0.015	*gehad*	0.612	0.016
bent	0.671	0.018	*werd*	0.510	0.023	*had*	0.454	0.021
geweest	0.460	0.045	*werden*	0.568	0.014	*hadden*	0.659	0.009
is	0.400	0.010	*word*	0.602	0.028	*heb*	0.592	0.022
waren	0.600	0.012	*worden*	0.417	0.016	*hebben*	0.402	0.015
was	0.498	0.028	*wordt*	0.586	0.025	*hebt*	0.602	0.030
zijn	0.415	0.022				*heeft*	0.459	0.028

Table C42

The cosine distances of all verb forms of komen, gaan and maken for the WIN9 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.771	0.062	*ga*	0.578	0.052	*gemaakt*	0.565	0.014
kom	0.748	0.030	*gaan*	0.774	0.048	*maak*	0.876	0.049
komen	0.517	0.037	*gaat*	0.644	0.043	*maakt*	0.642	0.062
komt	0.587	0.058	*gegaan*	0.657	0.035	*maakte*	0.645	0.027
kwam	0.598	0.023	*ging*	0.563	0.013	*maakten*	0.654	0.114
kwamen	0.651	0.048	*gingen*	0.762	0.073	*maken*	0.697	0.036

Table C43

The cosine distances of all verb forms of geven, doen and houden for the WIN9 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.687	0.033	*deden*	0.713	0.028	*gehouden*	0.630	0.038
gaven	0.630	0.017	*deed*	0.711	0.059	*hield*	0.611	0.040
geef	0.674	0.020	*doe*	0.615	0.028	*hielden*	0.673	0.040
geeft	0.685	0.024	*doen*	0.671	0.023	*hou*	0.770	0.018
gegeven	0.759	0.019	*doet*	0.803	0.031	*houd*	0.645	0.020
geven	0.600	0.024	*gedaan*	0.673	0.042	*houden*	0.652	0.028
						houdt	0.662	0.042

Table C44

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN9 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.535	0.035	*gezien*	0.747	0.024	*gekregen*	0.653	0.055
zeg	0.602	0.028	*zag*	0.758	0.059	*kreeg*	0.632	0.039
zeggen	0.523	0.024	*zagen*	0.861	0.061	*kregen*	0.666	0.074
zegt	0.587	0.035	*zie*	0.760	0.056	*krijg*	0.673	0.043
zei	0.575	0.024	*zien*	0.635	0.027	*krijgen*	0.668	0.015
zeiden	0.710	0.068	*ziet*	0.624	0.014	*krijgt*	0.665	0.044

Table C45

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN9 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.900	0.030	*brak*	0.796	0.040	*beweegt*	0.706	0.035
bieden	0.684	0.020	*braken*	0.731	0.026	*bewegen*	0.735	0.036
biedt	0.675	0.032	*breekt*	0.795	0.044	*bewogen*	0.754	0.043
boden	0.852	0.044	*breken*	0.693	0.070	*bewoog*	0.746	0.062
bood	0.822	0.024	*gebroken*	0.732	0.027			
geboden	0.812	0.029						

Table C46

The cosine distances of all verb forms of zijn, worden and hebben for the WIN10 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.608	0.030	*geworden*	0.678	0.012	*gehad*	0.667	0.031
bent	0.670	0.069	*werd*	0.430	0.042	*had*	0.574	0.012
geweest	0.529	0.018	*werden*	0.563	0.026	*hadden*	0.637	0.028
is	0.382	0.017	*word*	0.717	0.060	*heb*	0.463	0.046
waren	0.582	0.032	*worden*	0.475	0.036	*hebben*	0.390	0.017
was	0.427	0.032	*wordt*	0.491	0.025	*hebt*	0.660	0.049
zijn	0.411	0.035				*heeft*	0.404	0.043

Table C47

The cosine distances of all verb forms of komen, gaan and maken for the WIN10 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.754	0.048	*ga*	0.638	0.043	*gemaakt*	0.697	0.049
kom	0.629	0.028	*gaan*	0.700	0.084	*maak*	0.799	0.036
komen	0.573	0.028	*gaat*	0.622	0.040	*maakt*	0.745	0.025
komt	0.520	0.025	*gegaan*	0.689	0.066	*maakte*	0.640	0.019
kwam	0.614	0.018	*ging*	0.559	0.030	*maakten*	0.708	0.040
kwamen	0.533	0.046	*gingen*	0.762	0.015	*maken*	0.662	0.023

Table C48

The cosine distances of all verb forms of geven, doen and houden for the WIN10 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.590	0.062	*deden*	0.629	0.022	*gehouden*	0.616	0.044
gaven	0.627	0.031	*deed*	0.708	0.068	*hield*	0.639	0.041
geef	0.587	0.021	*doe*	0.472	0.020	*hielden*	0.772	0.045
geeft	0.690	0.079	*doen*	0.581	0.013	*hou*	0.737	0.021
gegeven	0.834	0.038	*doet*	0.715	0.039	*houd*	0.699	0.041
geven	0.586	0.029	*gedaan*	0.523	0.015	*houden*	0.584	0.038
						houdt	0.689	0.081

Table C49

The cosine distances of all verb forms of zeggen, zien and krijgen for the WIN10 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.587	0.013	*gezien*	0.832	0.028	*gekregen*	0.699	0.023
zeg	0.592	0.038	*zag*	0.676	0.078	*kreeg*	0.578	0.068
zeggen	0.528	0.034	*zagen*	0.797	0.083	*kregen*	0.658	0.030
zegt	0.552	0.063	*zie*	0.764	0.018	*krijg*	0.726	0.056
zei	0.531	0.037	*zien*	0.626	0.036	*krijgen*	0.670	0.035
zeiden	0.785	0.068	*ziet*	0.609	0.063	*krijgt*	0.726	0.024

Table C50

The cosine distances of all verb forms of bieden, breken and bewegen for the WIN10 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.900	0.047	*brak*	0.912	0.066	*beweegt*	0.898	0.044
bieden	0.674	0.030	*braken*	0.734	0.047	*bewegen*	0.769	0.052
biedt	0.613	0.044	*breekt*	0.778	0.047	*bewogen*	0.776	0.060
boden	0.827	0.028	*breken*	0.768	0.056	*bewoog*	0.733	0.052
bood	0.817	0.013	*gebroken*	0.804	0.049			
geboden	0.823	0.053						

Table C51

The cosine distances of all verb forms of zijn, worden and hebben for the DIM50 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.505	0.014	*geworden*	0.490	0.046	*gehad*	0.484	0.055
bent	0.400	0.041	*werd*	0.240	0.005	*had*	0.429	0.022
geweest	0.383	0.020	*werden*	0.444	0.016	*hadden*	0.422	0.011
is	0.243	0.020	*word*	0.476	0.041	*heb*	0.402	0.049
waren	0.376	0.027	*worden*	0.257	0.021	*hebben*	0.206	0.019
was	0.318	0.039	*wordt*	0.401	0.025	*hebt*	0.672	0.069
zijn	0.314	0.031				*heeft*	0.272	0.025

Table C52

The cosine distances of all verb forms of komen, gaan and maken for the DIM50 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.608	0.016	*ga*	0.439	0.046	*gemaakt*	0.532	0.044
kom	0.582	0.030	*gaan*	0.470	0.030	*maak*	0.677	0.024
komen	0.478	0.029	*gaat*	0.481	0.025	*maakt*	0.442	0.015
komt	0.327	0.049	*gegaan*	0.658	0.052	*maakte*	0.520	0.002
kwam	0.437	0.025	*ging*	0.318	0.030	*maakten*	0.547	0.034
kwamen	0.459	0.023	*gingen*	0.694	0.071	*maken*	0.396	0.020

Table C53

The cosine distances of all verb forms of geven, doen and houden for the DIM50 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.511	0.024	*deden*	0.512	0.043	*gehouden*	0.474	0.033
gaven	0.595	0.026	*deed*	0.609	0.065	*hield*	0.563	0.024
geef	0.512	0.027	*doe*	0.499	0.029	*hielden*	0.732	0.024
geeft	0.502	0.045	*doen*	0.388	0.021	*hou*	0.611	0.015
gegeven	0.676	0.048	*doet*	0.495	0.030	*houd*	0.616	0.032
geven	0.418	0.024	*gedaan*	0.684	0.018	*houden*	0.465	0.027
						houdt	0.659	0.046

Table C54

The cosine distances of all verb forms of zeggen, zien and krijgen for the DIM50 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.370	0.027	*gezien*	0.676	0.058	*gekregen*	0.621	0.041
zeg	0.560	0.031	*zag*	0.571	0.025	*kreeg*	0.563	0.035
zeggen	0.397	0.041	*zagen*	0.832	0.050	*kregen*	0.660	0.019
zegt	0.597	0.024	*zie*	0.615	0.017	*krijg*	0.587	0.047
zei	0.410	0.040	*zien*	0.478	0.031	*krijgen*	0.585	0.009
zeiden	0.794	0.043	*ziet*	0.620	0.034	*krijgt*	0.577	0.036

Table C55

The cosine distances of all verb forms of bieden, breken and bewegen for the DIM50 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.818	0.032	*brak*	0.640	0.022	*beweegt*	0.917	0.058
bieden	0.355	0.013	*braken*	0.827	0.038	*bewegen*	0.675	0.071
biedt	0.508	0.054	*breekt*	0.892	0.016	*bewogen*	0.666	0.030
boden	0.879	0.051	*breken*	0.527	0.041	*bewoog*	0.712	0.051
bood	0.815	0.066	*gebroken*	0.429	0.075			
geboden	0.632	0.022						

Table C56

The cosine distances of all verb forms of zijn, worden and hebben for the DIM200 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.615	0.027	*geworden*	0.732	0.038	*gehad*	0.672	0.021
bent	0.656	0.035	*werd*	0.607	0.026	*had*	0.720	0.032
geweest	0.712	0.023	*werden*	0.638	0.027	*hadden*	0.723	0.026
is	0.671	0.047	*word*	0.670	0.020	*heb*	0.621	0.012
waren	0.724	0.014	*worden*	0.600	0.016	*hebben*	0.583	0.020
was	0.692	0.011	*wordt*	0.654	0.035	*hebt*	0.702	0.037
zijn	0.613	0.028				*heeft*	0.662	0.018

Table C57

The cosine distances of all verb forms of komen, gaan and maken for the DIM200 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.718	0.027	*ga*	0.615	0.016	*gemaakt*	0.632	0.030
kom	0.699	0.034	*gaan*	0.678	0.029	*maak*	0.736	0.048
komen	0.697	0.032	*gaat*	0.635	0.032	*maakt*	0.769	0.063
komt	0.687	0.044	*gegaan*	0.712	0.015	*maakte*	0.631	0.034
kwam	0.591	0.023	*ging*	0.675	0.019	*maakten*	0.628	0.024
kwamen	0.597	0.042	*gingen*	0.694	0.049	*maken*	0.667	0.031

Table C58

The cosine distances of all verb forms of geven, doen and houden for the DIM200 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.669	0.026	*deden*	0.738	0.024	*gehouden*	0.744	0.013
gaven	0.735	0.045	*deed*	0.696	0.023	*hield*	0.698	0.039
geef	0.645	0.026	*doe*	0.654	0.023	*hielden*	0.783	0.011
geeft	0.745	0.030	*doen*	0.666	0.014	*hou*	0.695	0.031
gegeven	0.693	0.047	*doet*	0.640	0.045	*houd*	0.799	0.033
geven	0.659	0.030	*gedaan*	0.716	0.053	*houden*	0.570	0.062
						houdt	0.676	0.051

Table C59

The cosine distances of all verb forms of zeggen, zien and krijgen for the DIM200 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.721	0.068	*gezien*	0.736	0.025	*gekregen*	0.861	0.074
zeg	0.710	0.011	*zag*	0.754	0.046	*kreeg*	0.642	0.021
zeggen	0.719	0.024	*zagen*	0.854	0.031	*kregen*	0.697	0.036
zegt	0.698	0.033	*zie*	0.713	0.020	*krijg*	0.807	0.039
zei	0.623	0.014	*zien*	0.676	0.015	*krijgen*	0.665	0.091
zeiden	0.774	0.027	*ziet*	0.642	0.027	*krijgt*	0.731	0.045

Table C60

The cosine distances of all verb forms of bieden, breken and bewegen for the DIM200 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.707	0.036	*brak*	0.806	0.014	*beweegt*	0.751	0.029
bieden	0.674	0.021	*braken*	0.758	0.015	*bewegen*	0.733	0.019
biedt	0.698	0.025	*breekt*	0.874	0.027	*bewogen*	0.770	0.031
boden	0.842	0.023	*breken*	0.680	0.013	*bewoog*	0.763	0.027
bood	0.775	0.037	*gebroken*	0.782	0.032			
geboden	0.805	0.030						

Table C61

The cosine distances of all verb forms of zijn, worden and hebben for the DIM300 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.709	0.015	*geworden*	0.723	0.022	*gehad*	0.712	0.058
bent	0.714	0.048	*werd*	0.644	0.033	*had*	0.737	0.060
geweest	0.704	0.021	*werden*	0.691	0.019	*hadden*	0.703	0.020
is	0.665	0.035	*word*	0.784	0.023	*heb*	0.599	0.042
waren	0.726	0.013	*worden*	0.702	0.031	*hebben*	0.630	0.024
was	0.668	0.028	*wordt*	0.696	0.042	*hebt*	0.708	0.031
zijn	0.682	0.062				*heeft*	0.658	0.041

Table C62

The cosine distances of all verb forms of komen, gaan and maken for the DIM300 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.731	0.022	*ga*	0.661	0.037	*gemaakt*	0.692	0.018
kom	0.739	0.014	*gaan*	0.664	0.046	*maak*	0.759	0.034
komen	0.709	0.025	*gaat*	0.688	0.043	*maakt*	0.769	0.048
komt	0.703	0.029	*gegaan*	0.722	0.054	*maakte*	0.700	0.031
kwam	0.671	0.047	*ging*	0.737	0.034	*maakten*	0.711	0.049
kwamen	0.697	0.027	*gingen*	0.727	0.030	*maken*	0.709	0.028

Table C63

The cosine distances of all verb forms of geven, doen and houden for the DIM300 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.692	0.020	*deden*	0.739	0.019	*gehouden*	0.744	0.041
gaven	0.721	0.022	*deed*	0.688	0.013	*hield*	0.692	0.024
geef	0.684	0.040	*doe*	0.746	0.046	*hielden*	0.744	0.031
geeft	0.716	0.033	*doen*	0.704	0.046	*hou*	0.707	0.031
gegeven	0.759	0.055	*doet*	0.720	0.048	*houd*	0.762	0.017
geven	0.647	0.023	*gedaan*	0.683	0.042	*houden*	0.695	0.039
						houdt	0.718	0.032

Table C64

The cosine distances of all verb forms of zeggen, zien and krijgen for the DIM300 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.768	0.036	*gezien*	0.730	0.023	*gekregen*	0.773	0.030
zeg	0.743	0.036	*zag*	0.705	0.037	*kreeg*	0.684	0.068
zeggen	0.657	0.051	*zagen*	0.831	0.036	*kregen*	0.734	0.057
zegt	0.751	0.036	*zie*	0.763	0.027	*krijg*	0.834	0.024
zei	0.675	0.036	*zien*	0.708	0.037	*krijgen*	0.741	0.072
zeiden	0.744	0.041	*ziet*	0.700	0.029	*krijgt*	0.746	0.036

Table C65

The cosine distances of all verb forms of bieden, breken and bewegen for the DIM300 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.756	0.033	*brak*	0.821	0.023	*beweegt*	0.821	0.023
bieden	0.772	0.054	*braken*	0.779	0.059	*bewegen*	0.705	0.053
biedt	0.722	0.048	*breekt*	0.875	0.037	*bewogen*	0.785	0.031
boden	0.774	0.032	*breken*	0.725	0.027	*bewoog*	0.681	0.010
bood	0.816	0.017	*gebroken*	0.762	0.032			
geboden	0.735	0.032						

Table C66

The cosine distances of all verb forms of zijn, worden and hebben for the DIM400 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.692	0.051	*geworden*	0.715	0.018	*gehad*	0.732	0.034
bent	0.724	0.034	*werd*	0.732	0.015	*had*	0.714	0.063
geweest	0.735	0.029	*werden*	0.703	0.034	*hadden*	0.719	0.016
is	0.682	0.042	*word*	0.785	0.017	*heb*	0.598	0.029
waren	0.703	0.031	*worden*	0.743	0.012	*hebben*	0.658	0.049
was	0.756	0.025	*wordt*	0.702	0.038	*hebt*	0.753	0.019
zijn	0.749	0.049				*heeft*	0.709	0.042

Table C67

The cosine distances of all verb forms of komen, gaan and maken for the DIM400 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.711	0.030	*ga*	0.680	0.015	*gemaakt*	0.644	0.030
kom	0.763	0.037	*gaan*	0.789	0.034	*maak*	0.809	0.011
komen	0.722	0.035	*gaat*	0.677	0.028	*maakt*	0.744	0.034
komt	0.707	0.019	*gegaan*	0.776	0.030	*maakte*	0.713	0.048
kwam	0.619	0.009	*ging*	0.761	0.036	*maakten*	0.748	0.020
kwamen	0.698	0.030	*gingen*	0.759	0.026	*maken*	0.750	0.056

Table C68

The cosine distances of all verb forms of geven, doen and houden for the DIM400 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.771	0.038	*deden*	0.701	0.014	*gehouden*	0.692	0.030
gaven	0.740	0.035	*deed*	0.724	0.012	*hield*	0.764	0.027
geef	0.713	0.024	*doe*	0.703	0.033	*hielden*	0.746	0.032
geeft	0.726	0.018	*doen*	0.766	0.020	*hou*	0.670	0.034
gegeven	0.719	0.017	*doet*	0.701	0.062	*houd*	0.750	0.032
geven	0.648	0.027	*gedaan*	0.666	0.016	*houden*	0.710	0.007
						houdt	0.722	0.022

Table C69

The cosine distances of all verb forms of zeggen, zien and krijgen for the DIM400 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.693	0.021	*gezien*	0.776	0.017	*gekregen*	0.810	0.023
zeg	0.751	0.028	*zag*	0.718	0.025	*kreeg*	0.689	0.024
zeggen	0.746	0.010	*zagen*	0.816	0.024	*kregen*	0.748	0.022
zegt	0.748	0.018	*zie*	0.764	0.035	*krijg*	0.776	0.034
zei	0.687	0.021	*zien*	0.676	0.014	*krijgen*	0.745	0.024
zeiden	0.791	0.003	*ziet*	0.742	0.033	*krijgt*	0.717	0.025

Table C70

The cosine distances of all verb forms of bieden, breken and bewegen for the DIM400 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.788	0.038	*brak*	0.792	0.027	*beweegt*	0.742	0.022
bieden	0.731	0.019	*braken*	0.764	0.017	*bewegen*	0.743	0.026
biedt	0.708	0.036	*breekt*	0.861	0.022	*bewogen*	0.732	0.030
boden	0.757	0.014	*breken*	0.741	0.045	*bewoog*	0.695	0.010
bood	0.769	0.037	*gebroken*	0.836	0.015			
geboden	0.734	0.008						

Table C71

The cosine distances of all verb forms of zijn, worden and hebben for the DIM500 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.747	0.019	*geworden*	0.714	0.020	*gehad*	0.777	0.045
bent	0.689	0.018	*werd*	0.723	0.025	*had*	0.712	0.015
geweest	0.689	0.012	*werden*	0.696	0.042	*hadden*	0.666	0.016
is	0.663	0.037	*word*	0.774	0.030	*heb*	0.643	0.021
waren	0.733	0.024	*worden*	0.725	0.021	*hebben*	0.727	0.030
was	0.721	0.006	*wordt*	0.743	0.023	*hebt*	0.717	0.024
zijn	0.677	0.032				*heeft*	0.659	0.026

Table C72

The cosine distances of all verb forms of komen, gaan and maken for the DIM500 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.740	0.017	*ga*	0.707	0.029	*gemaakt*	0.664	0.028
kom	0.752	0.035	*gaan*	0.699	0.038	*maak*	0.806	0.020
komen	0.630	0.029	*gaat*	0.702	0.027	*maakt*	0.741	0.033
komt	0.679	0.021	*gegaan*	0.675	0.038	*maakte*	0.702	0.009
kwam	0.710	0.041	*ging*	0.731	0.023	*maakten*	0.720	0.010
kwamen	0.701	0.015	*gingen*	0.727	0.036	*maken*	0.701	0.035

Table C73

The cosine distances of all verb forms of geven, doen and houden for the DIM500 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.706	0.017	*deden*	0.683	0.042	*gehouden*	0.744	0.037
gaven	0.712	0.030	*deed*	0.659	0.018	*hield*	0.709	0.020
geef	0.688	0.027	*doe*	0.677	0.030	*hielden*	0.773	0.025
geeft	0.707	0.024	*doen*	0.699	0.030	*hou*	0.709	0.013
gegeven	0.697	0.027	*doet*	0.718	0.040	*houd*	0.747	0.031
geven	0.714	0.016	*gedaan*	0.632	0.017	*houden*	0.708	0.034
						houdt	0.722	0.048

Table C74

The cosine distances of all verb forms of zeggen, zien and krijgen for the DIM500 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.707	0.022	*gezien*	0.736	0.019	*gekregen*	0.804	0.023
zeg	0.737	0.020	*zag*	0.685	0.009	*kreeg*	0.712	0.016
zeggen	0.725	0.030	*zagen*	0.792	0.007	*kregen*	0.737	0.033
zegt	0.789	0.014	*zie*	0.767	0.013	*krijg*	0.827	0.023
zei	0.749	0.023	*zien*	0.706	0.014	*krijgen*	0.733	0.025
zeiden	0.749	0.016	*ziet*	0.732	0.026	*krijgt*	0.745	0.010

Table C75

The cosine distances of all verb forms of bieden, breken and bewegen for the DIM500 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.696	0.024	*brak*	0.818	0.026	*beweegt*	0.764	0.027
bieden	0.732	0.030	*braken*	0.770	0.016	*bewegen*	0.736	0.023
biedt	0.698	0.023	*breekt*	0.878	0.020	*bewogen*	0.763	0.010
boden	0.749	0.010	*breken*	0.749	0.026	*bewoog*	0.723	0.012
bood	0.788	0.031	*gebroken*	0.721	0.013			
geboden	0.782	0.017						

Table C76

The cosine distances of all verb forms of zijn, worden and hebben for the DIM600 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
ben	0.739	0.026	*geworden*	0.695	0.018	*gehad*	0.786	0.022
bent	0.717	0.028	*werd*	0.729	0.020	*had*	0.693	0.026
geweest	0.731	0.028	*werden*	0.636	0.023	*hadden*	0.708	0.015
is	0.680	0.023	*word*	0.752	0.023	*heb*	0.658	0.009
waren	0.679	0.021	*worden*	0.705	0.027	*hebben*	0.681	0.040
was	0.723	0.031	*wordt*	0.683	0.013	*hebt*	0.754	0.022
zijn	0.668	0.023				*heeft*	0.662	0.042

Table C77

The cosine distances of all verb forms of komen, gaan and maken for the DIM600 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gekomen	0.716	0.019	*ga*	0.675	0.019	*gemaakt*	0.710	0.013
kom	0.722	0.017	*gaan*	0.701	0.028	*maak*	0.763	0.022
komen	0.677	0.020	*gaat*	0.702	0.025	*maakt*	0.701	0.018
komt	0.703	0.017	*gegaan*	0.686	0.014	*maakte*	0.696	0.023
kwam	0.669	0.014	*ging*	0.739	0.033	*maakten*	0.704	0.017
kwamen	0.664	0.025	*gingen*	0.701	0.023	*maken*	0.648	0.014

Table C78

The cosine distances of all verb forms of geven, doen and houden for the DIM600 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gaf	0.721	0.028	*deden*	0.691	0.016	*gehouden*	0.769	0.014
gaven	0.720	0.023	*deed*	0.676	0.019	*hield*	0.656	0.034
geef	0.708	0.027	*doe*	0.712	0.020	*hielden*	0.748	0.017
geeft	0.709	0.034	*doen*	0.707	0.019	*hou*	0.714	0.018
gegeven	0.681	0.023	*doet*	0.683	0.013	*houd*	0.733	0.030
geven	0.672	0.027	*gedaan*	0.688	0.020	*houden*	0.647	0.021
						houdt	0.714	0.027

Table C79

The cosine distances of all verb forms of zeggen, zien and krijgen for the DIM600 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
gezegd	0.713	0.024	*gezien*	0.661	0.014	*gekregen*	0.774	0.027
zeg	0.756	0.030	*zag*	0.721	0.011	*kreeg*	0.682	0.022
zeggen	0.715	0.015	*zagen*	0.808	0.019	*kregen*	0.699	0.014
zegt	0.694	0.017	*zie*	0.771	0.024	*krijg*	0.772	0.017
zei	0.747	0.030	*zien*	0.685	0.010	*krijgen*	0.735	0.009
zeiden	0.747	0.035	*ziet*	0.664	0.023	*krijgt*	0.784	0.029

Table C80

The cosine distances of all verb forms of bieden, breken and bewegen for the DIM600 model

Verb form	Mean	SD	Verb form	Mean	SD	Verb form	Mean	SD
bied	0.697	0.029	*brak*	0.753	0.003	*beweegt*	0.762	0.018
bieden	0.726	0.013	*braken*	0.805	0.014	*bewegen*	0.746	0.015
biedt	0.715	0.042	*breekt*	0.849	0.024	*bewogen*	0.765	0.026
boden	0.747	0.026	*breken*	0.699	0.019	*bewoog*	0.737	0.007
bood	0.794	0.009	*gebroken*	0.725	0.026			
geboden	0.762	0.016						

Appendix D

The standard deviations of the most frequent verb lemmas per model

Table D1

The standard deviations of the most frequent verb lemmas per model for the WIN1-10 models

	Default	WIN1	WIN2	WIN3	WIN4	WIN6	WIN7	WIN8	WIN9	WIN10
zijn	0.034	0.038	0.039	0.029	0.025	0.021	0.025	0.022	0.022	0.033
worden	0.020	0.035	0.031	0.047	0.043	0.032	0.031	0.020	0.020	0.033
hebben	0.028	0.045	0.033	0.039	0.038	0.020	0.034	0.028	0.020	0.032
komen	0.022	0.025	0.029	0.050	0.030	0.034	0.028	0.021	0.043	0.032
gaan	0.032	0.027	0.021	0.046	0.044	0.041	0.028	0.027	0.044	0.046
maken	0.029	0.031	0.030	0.038	0.033	0.040	0.031	0.030	0.050	0.032
geven	0.026	0.033	0.048	0.051	0.038	0.023	0.027	0.033	0.023	0.043
doen	0.027	0.029	0.033	0.034	0.036	0.028	0.039	0.024	0.035	0.029
houden	0.033	0.028	0.030	0.043	0.034	0.032	0.024	0.038	0.032	0.044
zeggen	0.032	0.031	0.033	0.033	0.034	0.037	0.040	0.034	0.036	0.042
zien	0.030	0.049	0.036	0.042	0.049	0.026	0.042	0.022	0.040	0.051
krijgen	0.033	0.024	0.033	0.040	0.044	0.034	0.038	0.040	0.045	0.039
Mean	0.029	0.033	0.033	0.041	0.038	0.031	0.032	0.028	0.034	0.038

Table D2

The standard deviations of the most frequent verb lemmas per model for the DIM50-600 models

	Default	DIM50	DIM200	DIM300	DIM400	DIM500	DIM600
zijn	0.034	0.028	0.026	0.032	0.038	0.021	0.026
worden	0.020	0.026	0.027	0.028	0.022	0.027	0.021
hebben	0.028	0.036	0.024	0.039	0.036	0.025	0.025
komen	0.022	0.029	0.034	0.027	0.027	0.026	0.019
gaan	0.032	0.042	0.027	0.041	0.028	0.032	0.024
maken	0.029	0.023	0.038	0.035	0.033	0.023	0.018
geven	0.026	0.032	0.034	0.032	0.027	0.023	0.027
doen	0.027	0.034	0.030	0.036	0.026	0.029	0.018
houden	0.033	0.029	0.034	0.031	0.026	0.030	0.023
zeggen	0.032	0.034	0.030	0.039	0.017	0.021	0.025
zien	0.030	0.036	0.027	0.032	0.025	0.015	0.017
krijgen	0.033	0.031	0.051	0.048	0.025	0.022	0.020
Mean	0.029	0.032	0.032	0.035	0.028	0.024	0.022