Extract Similarities from Syntactic Contexts: a Distributional Semantic Model Based on Syntactic Distance

Alessandro Maisto*
Università degli Studi di Salerno

Distributional Semantics (DS) models are based on the idea that two words which appear in similar contexts, i.e. similar neighborhoods, have similar meanings. This concept was originally presented by Harris in his Distributional Hypothesis (DH) (Harris 1954). Even though DH forms the basis of the majority of DS models, Harris states in later works that only syntactic analysis can allow for a more precise formulation of the neighborhoods involved: the arguments and the operators.

In this work, we present a DS model based on the concept of Syntactic Distance inspired by a study of Harris's theories concerning the syntactic-semantic interface. In our model, the context of each word is derived from its dependency network generated by a parser. With this strategy, the co-occurring terms of a target word are calculated on the basis of their syntactic relations, which are also preserved in the event of syntactical transformations. The model, named Syntactic Distance as Word Window (SD-W2), has been tested on three state-of-the-art tasks: Semantic Distance, Synonymy and Single Word Priming, and compared with other classical DS models. In addition, the model has been subjected to a new test based on Operator-Argument selection. Although the results obtained by SD-W2 do not always reach those of modern contextualized models, they are often above average and, in many cases, they are comparable with the result of GLOVE or BERT.

1. Introduction

Distributional Semantics (DS) is a model of meaning whose theoretical foundation is the Distributional Hypothesis (DH). DH relies on the work of Harris (Harris 1954), which sets out the basis for a linguistic distributional methodology. The Distributional Hypothesis states that the statistical distribution of linguistic elements in context determines their semantic behavior (Lenci 2018).

In Distributional Semantics, the similarity between two words is calculated in terms of similarity between vectors. Word vectors describe the terms as a numerical representation of the various contexts in which they appear. Lenci (2018) reported two kinds of classification for DS models: the first regards the type of context, the latter the method of learning distributional vectors. Regarding the first classification, we can identify *region models*, in which the context of a word is the entire region the word appears in, and *word models*, which calculate context as a set of terms that appear at a certain distance from a target word. With reference to the first family of models, Ruge (1992) claims that the

* Università degli Studi di Salerno, Via Giovanni Paolo II, 139, Fisciano (SA) Italia. E-mail: amaisto@unisa.it.

larger the context, the larger the number of not semantically compatible terms included in the analysis. Moreover, Sahlgren (2008) considered the document as a context for a legacy of information retrieval. Since information retrieval is an artificial problem, "a document in the sense of a topical unit–unity is an artificial notion that hardly exists elsewhere" (Sahlgren 2008).

Word models, on the other hand, can be further divided into *window-based models* and *syntactic models*: the former consider a variable number of neighbor terms (the so-called "window") as the context of a given word. The latter seek to exploit syntactic dependency in order to obtain a more precise simulation of human knowledge-learning phenomena. However, considering the amount of pre-processing required, there is no empirical evidence for the supremacy of this kind of model (Sahlgren 2008).

In this paper, we aim to investigate the benefits of using syntactic information in Distributional Semantics, regardless of the amount of pre-processing required (this is not really a problem because of advances in syntactic parsing and machine performance, as well as the availability of ever-larger parsed corpora). We present a new syntactic model that benefits from a deeper reading of Harris's theories. We have based the new model on the concept of *syntactic distance* (Liu, Xu, and Liang 2017), the distance between a target word and other words syntactically connected to it, calculated by a dependency parser (Definition 1).

Definition 1
The Syntactic Distance is equivalent to the number of arcs of the dependency graph which separate two words.

All words at a certain syntactic distance from the target word may be included in the context of the target word. We have named our model the *Syntactic Distance as Word-window* (SD-W2) to highlight its use of the syntactic distance as a context-window selection metric.

The Distributional Hypothesis stated by Harris includes a level of syntactic analysis, which our model incorporates by taking a parsed corpus as input. The preliminary results show that our dependency-based system achieves results that are comparable to many other models and very close to the results of the BERT-based models.

The paper is structured as follows: in section 2, we analyze Harris's studies on the concept of "distribution", exploring the reasons why a syntactic model must be implemented. In section 3 we present a brief state of the art and point the focus on the most related works. In section 4, we present our methodology. In section 5 we present the experimental step. Finally, in section 6 we present the experiment outline and results.

2. The Distributional Hypothesis

Harris (1954) claimed that when someone speaks, they choose the next word from the members of those classes of words that usually occur in this position. Each language element can be grouped into classes, and while the relative occurrence of a class can be stated exactly, the occurrence of a particular member of one class relative to a particular member of another class must be calculated in terms of probability. In other words, given two linguistic elements A and B, if they "have almost identical environments", they can be considered synonyms (e.g., *oculist* and *eye-doctor*); if they have "some environments in common and some not" (e.g., *oculist* and *lawyer*), they have different meanings and this difference corresponds to the "amount of difference

of their environment" (Harris 1954, p. 157). The distributional structure reflects a sort of meaning structure in the way that "difference of meaning correlates with the difference of distribution" (Harris 1954, p. 156). The operation that studies the distributional structure is distributional analysis.

Distributional analysis is a basic process that Harris describes as being related to five distributional facts: a) possibility of *segmenting flows of speech* into parts (elements) to find regularities in the occurrence of one part relative to others; b) *similarity*, considered as the property of some elements to group with similar elements into sets; c) *dependence* of the elements in a group of similar objects on elements in another group; d) *substitutability* of elements that have the same environment; e) *domain*, such as the word, the phrase, the clause, in which both dependence and substitutability work. The distributional analysis output is a set of substitution classes or equivalence classes (Harris 1946, 1952).

Many authors have adopted the distributional hypothesis and the correlation between distribution and meaning for practical tasks: the first authors to exploit distributional analysis in a computational task were Schutze and colleagues (Schutze 1992a; Schütze 1992b; Schutze and Pedersen 1995). He presented a paper on word sense disambiguation based on a vector representation of word similarity derived from lexical co-occurrence. Subsequently, Landauer and Dumais (1997) proposed a model for the simulation of knowledge-learning phenomena based on local co-occurrence data in a large representative corpus, called Latent Semantic Analysis (LSA). Lund and Burgess (1996) introduced Hyperspace Analogue to Language (HAL), an algorithm that calculates the semantic similarity between two words by comparing the co-occurrence vectors of the two words with a Euclidean measure of distance. These approaches paved the way for the success of Distributional Semantics (DS).

Although early Distributional Semantics models display evidence of the influence of Harris's hypothesis, the distributional hypothesis is not explicitly mentioned as their theoretical foundation. Only later was the Distributional Hypothesis adopted by DS authors as a type of *a posteriori* justification for their work. Indeed, the above studies did not take into account some fundamental aspects of Harris's theories, such as the influence of syntax on the formulation of the neighborhoods of a word and the problem of non-contiguous elements of syntactic structures.

2.1 Syntax and Semantics in Harris

In Harris (1968, p. 209) there is an essential specification on the Distributional Hypothesis:

> *"...difference in meaning between words correlates with difference between them in respect to their word neighborhoods. Transformational analysis permits a more precise formulation of the neighborhoods involved: they are the arguments and the operators."*

The correlation between a word's neighborhoods and its syntactic context appears even more clearly in Harris's later works. In Harris (1988, 1991), he described language structure in terms of constraints. Each word combination is characterized by a set of constraints, "each of which precludes particular classes of combination from occurring in utterances of given language" (Harris 1991, p. 53). These constraints (*partial order*, *likelihood*, and *reduction*) act on the product of another constraint in a cascading mechanism.

The first constraint regards the *partial ordering on words* understood as "what gives a word-sequence the capacity to express fixed semantic relations among its words"

(Harris 1991, p. 5). It acts above the other constraints. It is the "essential one" (Harris 1991, p. 7) because it creates sentences.

In the partial order or Operator-Argument constraint, a word serves as the *Operator* over the other words called *Arguments*. The words of a language obtain their ability to co-occur in sentences thanks to the partial order: a word like *eat* is higher than *sheep* or *grass* because it can operate on nouns as in "*sheep eat grass*". Other operators such as *know* or *probable* are higher than eat because they can operate over it as in "*I know that sheep eat grass*". Sentences can be defined as word-sequences that satisfy this partial order. The operator-argument relations yield the meaning of the entire sentence by applying partial order relations to the meaning of the words. The sentence meaning is "the hierarchy of predicatings among the meanings of the words of the sentence" (Harris 1991, p. 8).

The likelihood constraint regards the meaning of words. For each argument word, there are some words that "are more likely than others to appear as operator on it" (Harris 1991, p. 5). In other words, the meaning of a word is determined by the selection of words (word-choice) that are operators of arguments in a given sentence (Harris 1976a, p. 263).

This constraint is strongly related to distributional analysis and the concept of *dependence*. *Dependence* is conceived as "a relation between a word and an ordered set of word classes". As exemplified by Harris (1991, p. 55), in "*the child sleeps*", the verb *sleep* depends on a word of a particular class of objects such as *Mary*, *John*, *the child*, etc. Therefore, as an argument of *sleep*, we can find a particular set of elements that corresponds to the set of Nouns. This dependency produces a *similarity* between the elements in the group. The dependence is never complete, but there are "various degrees and types of occurrence-dependence" (Harris 1954, p. 159). Among these nouns, we can find *John*, *the child*, *the dog*, and, more rarely, *the city* (*The city sleeps*), *the tree* (*Trees have to sleep each winter*). The *likelihood-gradation* between operators and arguments is a crucial relationship in language structure, and these inequalities in likelihood are not modified by transformations (Harris 1976b, p. 243).

The third constraint concerns the *reduction* of a word-sequence that helps produce more compact sentences. Certain words with a high likelihood contribute to the meaning of the sentence with a small amount of information (Harris 1991, p. 84). For example, the sequence *to come* in sentences like "*John expects Mary to come*", has a very high likelihood for the operator expects, and its reduction produces an acceptable sentence (*John expects Mary*). Harris identifies three kinds of widespread reductions. Reduction to zero (zeroing), which is the case of the example above. Reduction to affixes, as in the word *childhood*, in which the suffix *-hood* derives from the Old English *had*, "*state, condition*". Reduction to pronouns as in the sentence "*I met John, who sends regards*", which is a reduction from "*I met John; John – the preceding word has the same referent as the word before – sends regards*". "*John – the preceding word has the same referent as the word before*" is reduced to who, and, in some cases, can be zeroed ("*The money which is needed is unavailable*", "*The money needed is unavailable*") (Harris 1991, p. 81-82).

As indicated above, Harris states that each constraint acts on the product of another constraint; thus, the third constraint, reduction, acts on the product of the Likelihood constraint. The latter, in turn, acts on the product of the partial-order constraint. Hence, as affirmed by Harris, "given the meanings of the words, finding the operator- argument relations among the words of a sentence yields its meaning directly: that meaning is the hierarchy of predicatings among the meanings of the words of the sentence"(Harris 1991, p. 8). In other words, "the syntax of a sentence indicates its semantics" (Harris 1991, p. 9).

Reduction is included in the set of basic transformations (Harris 1991, p. 210). Those

basic transformations (zeroing or reduction, permutation of word-classes, single-word adjuncts, sentence nominalization, and conjoined sentences) make it possible to derive the *base sentence* (or kernel sentence) from two kinds of paraphrastic sentence: sentences with additional words (e.g. *the sheep eat grass*; *I know sheep eat grass*) and sentences with no addition but with a change (e.g. *He reads all day*; *He reads things all day*).
In all these transformations, the partial-order and the "major elements of meaning" are preserved (Harris 1991, p. 290). The word-sequence (given by the partial order) of unreduced sentences is not modified by reduction, and word-choice (resulting from likelihood and partial order constraints) is preserved under transformations. "With the preservation of word-choice comes meaning-preservation" (Harris 1991, p. 229).

These constraints suggest that, in Harris, the syntactic relation between operators and arguments yields the semantics of the sentence. Besides, the meaning of a single word depends on the likelihood that it will appear in its various operator-argument statuses. Reductions and transformations alter neither the operator-argument relation nor the likelihood inequalities.

Since a speech event is always developed in a single dimension of time, it needs a linear order that differs from the partial order. In addition to the three constraints illustrated above, Harris (1991, p. 6) hypothesizes that, after the partial order, the "words are put in one or more linear forms".

In another paper (Harris 1968), the author affirmed that one of the relevant properties of language is the *linear order* of entities. Though operators and their operand (argument)[1] must be contiguous, Harris contemplates that "later operators on the resultant may intervene between the earlier operator and its operand, separating them" (Harris 1968, p. 16). Thus, contiguity does not refer to single words but to well-formed subsequences that constitute the sentence. The construction of the sentence, stated Harris, must be formulated on the basis of entities that are larger than words "in respect to which there are no noncontiguous phenomena" (Harris 1968, p. 32).

2.1.1 How the SD-W2 model reflects Harris's constraints

Most DS models consider the context in its linear form when they find co-occurrences of a word. In fact, texts reflect in space the linearity of the temporal dimension in which speech is developed. However, this linear representation of a sentence does not reflect its structure, which must be described in terms of grammatical relations. By exploiting the syntactic relations emerging from a syntactic parsing process, the SD-W2 model aims to consider the three constraints mentioned above as a guideline to extract the context of words. Sentence 1 points out the differences between the two kinds of approach.

Example 1
The man who came into the bank with the gun and the mask shot the policeman.

According to Harris, Example 1 results from a set of transformation and reduction (mainly reduction to pronoun, zeroing, and conjunction) over a set of kernel sentences, each of which observes a specific partial order. The set of kernel sentences is as follows:

1. the man shot the policeman
2. someone came into the bank

1 Harris alternates between operands and arguments

3. someone had a gun
4. someone had a mask

As indicated above, transformations and reductions do not alter the partial order, so the information yielded by the kernel sentences must be preserved in Example 1.
Classical word-window models such as HAL or COALS consider windows of 4-10 words as being context linear. They produce co-occurrence values based on the linear distance between words. In Sentence 1, for example, a five word-window selects the sequence *who came into the bank* as the context of the subject *man*. They cannot even relate the subject *man* and the operator *shot* because, in Example 1, the distance between the subject and the main operator exceeds the window size.
Unlike classical word-window models, SD-W2 reflects the original structure given by the partial order in the four kernel sentences. Considering that *someone* in 2, 3, and 4 refers to the *man* in 1, the syntactic context of the four kernel sentences in terms of syntactic distance is the same. We have distance 1 between arguments (subject and complement) and the operator and distance 2 between the subject and the complement. The model can correctly connect the argument and its operators even if they are not contiguous or if a large relative clause separates them.

Table 1
Linear and Syntactic distance between the word *man* and the other nouns of the Example 1

	came	**bank**	**gun**	**shot**	**policeman**
Linear Distance	2	5	8	12	14
Syntactic Distance	1	2	2	1	2

Table 1 shows the linear and the syntactic distances between the noun *man* and the Verbs and Nouns in the sentence. The verb shot is 12 words away from the subject and cannot be included in the context of the noun by a 5 or 10 word-window. Our model captures this relationship in the same way that it captures the relation between *man* and the verb of the relative clause, *came*.

In addition, if the sentence were subject to additional transformations (*the policeman was shot by the man who came into the bank with the gun and the mask*), the distances remain unaltered, and the context of the word *man* is preserved.
Our model takes advantage of the dependencies between the words in the sentence that emerge from the automatic parsing in order to consider non-linear relations in the context selection. In this way, we can easily relate the operator with all its arguments, even if they are non-adjacent or represented by a pronoun. Only a model with these characteristics can capture the semantic structure of the sentence because its meaning depends on both syntax and semantics and the relation between them. A distributional semantics model cannot consider the sentence as a linear concatenation of elements because the semantic structure that underlies the syntactic structures is not linear. The context of a word must be considered as its partial order and must remain unaltered after reduction or transformation.

Since the 1990s, a relative small number of dependency-based models have been presented, (Padó and Lapata 2007; Grefenstette 1992; Lin 1997; Strzalkowski 1994). These models seek to exploit syntactic dependency so as to obtain a more precise simulation of Human knowledge-learning phenomena. There is no empirical evidence

for the supremacy of this kind of model in general tasks (Kiela and Clark (2014), and Lapesa and Evert (2017) reports substantially comparable results). In addition, syntactic models generally require a large amount of pre-processing. Nevertheless, thanks to improvements in syntactic parsers and computing power, we feel that using syntactic data to perform similarity computation is of primary importance.

In the next section, we will provide a rapid overview of the DS models that have most influenced our work.

3. Related Works

In section 2, we analyzed Harris's theories on meaning and the relation between syntax and semantics and how he directly or indirectly influences later theories.

Harris's distributional hypothesis is rooted in structuralist theories and in Saussure's concept of *valeur* (Sahlgren 2008, p. 5). The differential view of meaning that characterized Harris and, earlier, Bloomfield is based on the idea that signs are identified by their functional differences (the *sign's valeur*). A sign assumes a *valeur* by virtue of its "being different from other signs"; it therefore emerges only in a system and cannot exist in isolation. Saussure considered two kinds of relation in which functional differences emerge. *Syntagmatic* relations concern connections between words that co-occur (*in praesentia*); *paradigmatic* relations concern substitution, and related words that do not co-occur (*in absentia*).
According to this difference, Sahlgren (2008) classified distributional models as Syntagmatic or Paradigmatic models.

The first family of models focuses on Sentence Meaning. These models study polysemy, disambiguation, and semantic compositionality from a distributional point of view. Disambiguating polysemous words cannot be addressed with a traditional approach based on formal semantics, such as the standard Distributional Semantics Models (Baroni, Bernardi, and Zamparelli 2014). There are two predominant approaches: the first encodes all relevant information for a given word and then uses context to find the right meaning. The second builds different vectors for each word sense (Boleda 2020).

Related to the concept of paradigmatic and syntagmatic relations is the classification of first-, second- and third-order techniques produced by Grefenstette (1994). The author defines first-order techniques as those that look at the local context to discover what other words can be found among the neighbors of a given word. Second-order techniques look for terms that share the same environments. Third-order techniques create semantic groups of similar words by manipulating the list of similar words produced by a second-order technique.

Distributional Semantics Algorithms based on Harris's distributional hypothesis can be classified in the second family of models, paradigmatic models, or second and third-order techniques.
As pointed out in section 1, these models can be classified by using different criteria (Lenci 2018): if we consider the context selection, we can classify them into Word-Based models and Document-Based models. While document-based models consider a whole document as the context, word-based models take a variable number of words.

In the last few years, several models based on neural network algorithms have appeared. Since the introduction of Word2Vec (Mikolov et al. 2013b), these so called *predict models* (Baroni, Dinu, and Kruszewski 2014), have demonstrated their superiority over traditional models.

More recently, deep neural networks have been applied to traditional and predict models in order to overcome the idea that each token must correspond to a vector

(Peters et al. 2018): these latest-generation models represent a word with a number of vectors equivalent to the different sentence contexts in which it appears. For this reason, these models are called *contextualized word embeddings*.
Contextualized models work by learning the vectors as a function of internal states of a pre-trained encoder (Chersoni et al. 2021) such as Long Short Term Memory (LSTM) for feature-based approaches (Peters et al. 2018), or Transformers for fine-tuning approaches (Devlin et al. 2019). In particular, BERT (Devlin et al. 2019) and ELMo (Peters et al. 2018), became very popular in the last years because offers generalized solution to many computational linguistic tasks with very high performances.

The model proposed in this paper does not take this kind of technology into consideration. We aim to demonstrate that the influence of syntax on the generation of semantic word matrices could improve the results of DS models, regardless of the family the model belongs to.

As was illustrated in section 2, a large part of models consider words that belong to the same document or sentence as co-occurring. These models do not make use of linguistic data. However, many other models are built in such a way that linguistic knowledge affects the collection of distributional information. These models aim to use part of speech tags, lemmas, or dependencies. Since the proposed model is a word-based dependency model that explores paradigmatic relations, we will present a rapid overview of Distributional Semantics algorithms that influence our work.

3.1 Window-Based Models

Our overview begins with Hyperspace Analogue to Language (HAL) (Lund and Burgess 1996), which is considered one of the most influential Distributional models (Lenci 2008).

In HAL, the semantic similarity between two words is calculated by comparing word-vectors with Euclidean distance measures, extracted from a large co-occurrence matrix. HAL reads the corpus through an n-words window to generate the co-occurrence matrix. The window size suggested by the authors ranges between 5 and 10 words, and the corpus must include a large set of heterogeneous texts.

The authors use a lexicon of the 70,000 most frequently used terms of English to generate a HAL matrix with a dimension of 70,000 X 70,000 (Burgess 1998). Each word vector is processed with a multidimensional scaling algorithm to transform it into a bi-dimensional pictorial representation of the word. This procedure generates semantic knowledge by grouping semantic neighbors and grammatical knowledge. The corpus used to generate the matrix is 300 million words of English text from Usenet newsgroups. This methodology makes it possible to represent the semantic meaning of words and bring out the characterization of a variety of aspects of lexical ambiguity (Burgess 2001). HAL exerted a major influence on many later models (Audet and Burgess 1999; Azzopardi, Girolami, and Crowe 2005; Rohde, Gonnerman, and Plaut 2006).

In particular, *Correlated Occurrence Analogue to Lexical Semantics* (COALS) (Rohde, Gonnerman, and Plaut 2006) achieves considerably better performance levels. In HAL, the authors believe that high-frequency columns make an excessive contribution to the distance measure. COALS employs a normalization strategy that solves this issue. The model is set on a flat 4-word window and computed on the 100,000 most frequent words as columns and 1 million rows. Once the 4-word window completes the matrix building process, the co-occurrence value is replaced with a value calculated as a Pearson Correlation between each row. The Pearson Correlation measures the linear dependence between two variables. It is one of the first measures of correlation and remains one of

the most widely used measures of relationship (Schober, Boer, and Schwarte 2018). The Pearson Correlation generates values in a -1 to 1 range, in which -1 is a total negative correlation, 1 is a total positive correlation, and 0 represents the complete absence of correlation. The authors transform all negative values into 0 and square all other values. By setting all negative values to 0, the authors obtain a scattered matrix, losing information on anti-correlated words that do not generate similarity values between words. Conversely, by squaring all positive values, the importance of many small values is exalted in comparison to the few larger ones.

As regards vector length, the authors choose to eliminate purely syntactic words such as determiners or punctuation symbols, using a 14,000 columns matrix. Finally, vector similarity is calculated by using the Pearson Correlation once again.

The model was tested on several tasks, including word-pair similarity ratings, multiple-choice vocabulary tests, yielding a better performance than other state-of-the-art models. The results were also confirmed in Jurgens and Stevens (2010), who compare different algorithms.

A different Window-Based family of models employs a Random Indexing approach (Kanerva, Kristoferson, and Holst 2000). Random Indexing produces low-dimensional random vector representations of each context. When the word-window scans the corpus, each time a word occurs in a context, the random vector is added to the context vector (Sahlgren 2005). Since the dimensionality of the random vector is reduced, the context vectors will also have the same dimension. This method makes it possible to build the matrix incrementally, with low-dimension and with any kind of context selection method.

Lapesa and Evert (2014) investigated the impact of various Word-window model parameters on a number of traditional semantic tasks. Three parameters appear to have a particularly significant impact on a model's performance: *score* (how the algorithm assigns a co-occurrence value to the words in the word-window), *transformation* (how the co-occurrence scores are then transformed so as to reduce the features' asymmetry) and *distance metric*.

The impact of those parameters, and in particular of *transformation* can explain the better performance of COALS compared to HAL: since the other parameters are similar for both models, the introduction of a matrix transformation is the primary distinction between them. While HAL does not provide any kind of transformation of the matrix, COALS employs Pearson's transformation.

Other parameters (*corpus*, *window size*, *dimensionality reduction*) also exerted an influence, but they varied more widely in response to the task. For example, the *Difference of Means* between reduced and unreduced models is quite substantial for the TOEFL task; for the other tasks, the use of the WaCkypedia corpus (Baroni et al. 2009) yields better results.

3.2 Dependency-Based Models

Dependency-Based Distributional Semantics, also known as syntax-based distributional semantics, inspires a class of algorithms that use linguistic annotation to improve the results of similarity measure extraction. In general, we can consider these models as belonging to word-based models because only words belonging to the same sentence are included in the context. Unlike HAL, this kind of method does not assign co-occurrence values according to nearness between words, but they take advantage of the syntactic relations shown by a syntactic parser.

Regardless of the amount of pre-processing required, the differences between syntactic models and word-window models in terms of performance are difficult to judge. Traditional Word-window models, also known as bag-of-words models, generally achieve the best performance in classification tasks, while *bag-of-arguments* models (Dependency models) perform better in predicting argument expectations (Chersoni et al. 2017). Levy and Goldberg (2014) train the word-window model *SkipGram* (Mikolov et al. 2013b) and perform their experiments with a dependency-based context. They show that the dependency-based context yields a different embedding, such as *functional similarities of a cohyponym nature*.

The first dependency-based algorithm to return promising results in distributional semantics was presented by Grefenstette (1992). The paper's idea was to take advantage of the growing availability of syntactic parsers to select the syntactic context of words. The model, called *Sextant*, derives similarity measures that consider the overlapping of all contexts associated with a target word over the corpus.

Other influential syntactic models were presented by Strzalkowski (1994) and Lin (1997): Strzalkowski (1994) presents a dependency-based methodology included within an information retrieval task. The authors propose the extraction of a set of head+modifier pairs from a parsed text, which are used as occurrence contexts for each term included in them. Two terms that share some modifiers but appear in a few distinct contexts receive a similarity coefficient of between 0 and 1. Lin (1997) proposes a Word Sense Disambiguation algorithm based on a Similarity Measure calculated through a syntactic context. The local context of a word is defined as a triple of dependency relations in which the word is the head or the modifier. The authors construct Local Context Databases by extracting this kind of relation and using word frequency and the likelihood ratio to give a distance value. Each target word is described as a triple (type, word, position) and a set of word-frequency-likelihood-triples.

Inspired by the works of Lin and Strzalkowski, Padó and Lapata (2007) developed a model based on the notion of *paths*. Paths are sequences of dependency edges that connect two words, the use of which makes it possible to represent both direct and indirect relationships between words. There are three new parameters related to paths: the *Context selection function* determines which path in the dependency graph contributes to the representation of the target word; the *path value function* assigns weights to paths, for instances, giving more weight to paths containing subjects and objects; the *basis mapping function* establishes the size of the semantic space. In their work, the authors list three different context selection functions, minimum, medium, and maximum, respectively of length 1, length <=3 and length <=4, and three path value functions: plain, which assigns 1 to every path, length, which assigns a value inversely proportional to the length of the path and gram-rel, which ranks paths by using a value that reflects the salience of their grammatical relations (i.e., subjects are more salient than objects). The authors also define an *optimal dependency-based model* which uses the medium context selection function and the length path value function, with 2000 basis elements. They train the model on the *British National Corpus* (100 million words) and test it on three tasks: Single-word Priming, Detection of Synonymy, and Sense Ranking. The model achieves performance levels comparable to or higher than state-of-the-art models in all the selected tasks.

More recently, Baroni and Lenci (2010) proposed an approach called *Distributional Memory* in which the authors seek to solve the problem of building a different distributional model for each different semantic task. The methodology adopted entails the extraction of co-occurrence as a ternary geometrical object of the kind word-link-word, called the third-order tensor. The tuple word-link-word is made up of two content

words and a syntagmatic co-occurrence link between them: for example, the tuple <*marine,use,bomb*> denotes that the word *marine* co-occurs with the word *bomb*, with the word *use* representing the syntagmatic link between the two.

Distributional Memory provide two different models: the *dependency model* uses a set of links for *noun-verb*, *noun-noun* and *adjective-noun* pairs, which includes *verbs* (*the soldier is reading a book* → <*soldier,* verb*, book*>), the *subject of intransitive verbs* (*the teacher is singing* → <*teacher,* sbj_intr*, sing*>), the *noun modifier* (*good teacher* → <*good,* nmod*, teacher*>), etc.

The *lexical model* includes complex links, which take into account the morphological features of the pair words: *POS*, *number*, *tense*, *presence of articles*, *adjectives*, *adverbial modifier*, *auxiliary* or *modal verbs*. For example, the sentence *The tall soldier has already shot* is represented by the tuple <*soldier,* sbj_intr+n-the-j+vn-aux-already, *shot*>. The suffix of the link shows that the first word (*soldier*) is a singular noun (*n*), definite (*the*) and has an adjective (*j*), and that the second word (*shot*) is a past-participle (*vn*) with an auxiliary (*aux*) and is modified by an adverb (*already*).

Subsequently, matrices are generated directly from the tensor to perform a specific semantic task in a defined space. The model was tested on different semantic tasks and achieved a performance that, in some cases, was slightly lower than other models constructed ad hoc for the task. Nevertheless, the advantage of using a single general model that does not need to be retrained for each new task compensates for the lower performance.

Dependency-based models have also been tested on a variety of tasks to understand how different parameters affect their performance Lapesa and Evert (2017). The Dependency-based models work similarly to the window-based models in terms of performance and best values for a significant number of parameters (metric, score, transformation).

4. The SD-W2 algorithm

In order to perform a distributional analysis and calculate the similarity values from the context of the words, we choose to include a level of syntactic analysis in our model. This makes it possible to draw the real connections between words and overcome the linear vision of the sentence adopted by the word-window models.

These models extract similarity among words by calculating the similarity of their likelihood: if two words appear near the same group of words (i.e. they have similar contexts) in large corpora, then they have similar meanings. Word-based models calculate the context as a connection value between a word and all the words immediately adjacent to the target word or within a certain distance from it.

Nevertheless, as highlighted in section 2, Harris explicitly states that analysis of the meaning must rely on the first constraint (partial order). The partial order constraint acts over different hierarchies of linguistic elements: at the higher level, it works on operators that act over lower operators (i.e the verb *said*, which acts over other operators such as *eat* in sentences like "*I said that sheep eat grass*"); it acts on operator-argument relations (i.e. the verb *sleep* and its argument *child* in "*the child sleeps*"); but there also exists a hierarchical relation between the noun *reading* and the noun *book* in a sequence like *the reading of the book*. Harris (1957), assumed that the sentence "*the reading of the book is fast*" results from a set of transformations over two kernel sentences:

- k_1: the reading is fast

- k_2: someone read the book

The transformation involved are the following:

- $S \leftrightarrow N$ of k_2: the reading of the book by someone
- k_1 overlap with k_2: the reading of the book by someone is fast
- zeroing of "by someone": the reading of the book is fast

In other words, the k_2 kernel is nominalized (from *to read* to *the reading of*) and is overlapped with k_1. Finally, reduction allows the sequence *by someone* to be deleted because it brings a very small amount of information (there is always someone that reads a book).

Besides, reduction and transformations hide elements that appear with high frequency values in specific contexts and change the shape of a sentence, leaving syntactic relations unaltered. In this way, *reading* and *book* were also involved in an operator-argument relation and must be taken into account in the semantic analysis of the sentence. At a syntactic level, the distance between *reading* and *book* in the final sentence corresponds to the distance between *read* and *book* in k_2. It is the nature of the relationship that has changed.

Based on these assumptions, the proposed model attempts to extract co-occurrence values by considering the syntactic connections between words, regardless of typology or direction. The underlying idea is that the *syntactic context* of a word can be calculated on a parsed text by considering a measure derived from the concept of Syntactic Distance (Liu, Xu, and Liang 2017). As a quantified value, it works as a word-window that scrolls the text, not in its linear order but in its syntactic *partial order*. In exactly the same way as other models, the syntactic distance is converted into a numerical value which propagates through the network of relations described by the parsed text as shown in figure 1.

As illustrated by the figure, the distance is equal to the number of nodes in the syntactic sentence graph separating the target word from the other words in the sentence. At each distance, there may appear as many words as there are incoming and outgoing connections for a node.

4.1 Description of the Algorithm

The algorithm relies on the input of three external elements:

1. a base-dictionary that includes all the terms for which a vector representation is sought. We used a non-flexed dictionary and each vector will represent a single Lemma;
2. a dimension-dictionary that includes the terms representing the dimensions of each vector, i.e the columns of the matrix. This dictionary must also contain non-flexed terms;
3. a collection of documents in CoNLL format. CoNLL (Buchholz and Marsi 2006) provides a great deal of linguistic information about the text in table form. The rows of CoNLL tables represent the words that make up the document. The columns include an ID number, the FORM or token, LEMMA, universal POS Tags, HEAD, which indicates the ID of the

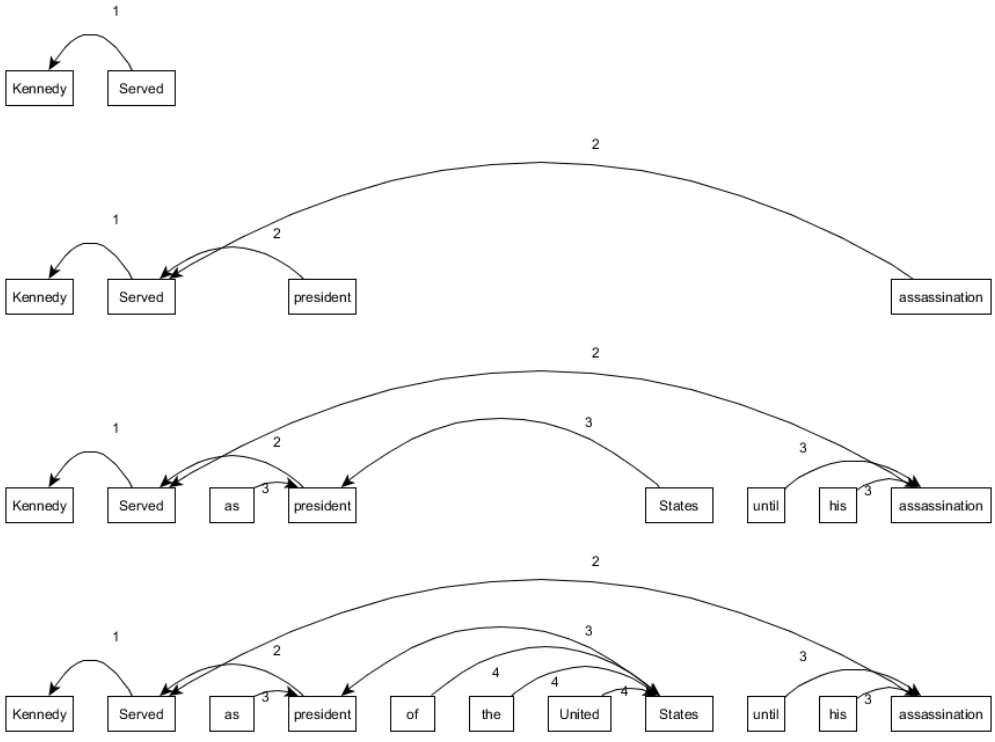


Figure 1
Syntactic Distance values for the word *Kennedy* in the sentence "Kennedy served as president of the United States until his assassination"

headword, and DEPREL, which indicates the nature of the dependency relation.

The algorithm proceeds by mapping the two dictionaries with a number that corresponds to the column/row of the matrix. Then, the algorithm takes into consideration a single sentence.

Algorithm 1 reports a description of the method that converts the input CoNLL sentence into a Sentence Graph Structure (SGS).

Algorithm 1 generation of the SGS from the CoNLL Sentence

Input: conllSentence
1: **for** line in conllSentence **do**
2: add [line(ID),line(HEAD)] to SGS
3: add [line(HEAD),line(ID)] to SGS
4: **end for**
Output: SGS

SGS is an edge graph in which the connections are represented by two values: the ID (source) and the HEAD (target) of each CoNLL row. In the SGS, connections

have no direction because the syntactic distance is calculated for each pair of connected elements, regardless of the nature and direction of that connection. In order to represent the bi-directionality of the SGS simply, the algorithm also inserts the inverse connection of each arc into the graph structure.

By way of an example, we consider the sentence in CoNLL format of table 2.

Table 2
Parser data in CoNNL format of the sentence "Kennedy served as president of the United States until his assassination"

ID	FORM	LEMMA	POS	HEAD	DEPREL
1	Kennedy	Kennedy	NNP	2	nsubj
2	served	serve	VBD	0	ROOT
3	as	as	IN	4	case
4	president	president	NN	2	obl
5	of	of	IN	8	case
6	the	the	DT	8	det
7	United	united	NNP	8	compound
8	States	states	NNP	4	nmod
9	until	until	IN	11	case
10	his	his	PRP$	11	nmod:poss
11	assassination	assassination	NN	2	obl

The word *president* points to the word *Kennedy* and, consequently, they are considered to have distance 1; but *president* also has distance 1 with the words that point to it (*as*, *state*). From the word *president*, whose ID number is 4, generates the arc {4,2}; since the connections are bidirectional, it also generates {2,4}. In addition, the words that point to *president* generate the arcs {3,4}, {4,3}, {8,4}, and {4,8}.

The SGS of the sentence in tab 4.1 will includes the following list of edges:

{0,2};{1,2};{2,0};{2,1};{2,4};{2,11};{3,4};{4,2};{4,3};{5,8};{6,8};
{7,8};{8,4};{8,5};{8,6};{8,7};{9,11};{10,11};{11,2};{11,9};{11,10}

Once the algorithm has processed the dictionaries and created the SGS, it removes all the SGS edges that involve the ROOT (all the pairs that include a zero). At this point, the algorithm starts the syntactic context analysis by inserting the sentence co-occurrence values into the matrix. Algorithm 2 describes the syntactic co-occurrence analysis.

The loop takes as input the SGS, the target word and a structure that maps the ID of each word with its POS and Lemma. It also needs two parameters:

- **Syntactic Distance**: the variable *windowSize* corresponds to the size of the syntactic window taken into account. This value ranges from 1 to 5.
- **Weighting function**: the function that determines the weight to assign to co-occurring words according to their distance.

In the first part of algorithm 2, it extracts the words directly connected with the target word. It assigns the value 1 to the connected words and stores their ID values

Algorithm 2 SyntCoOccAnalysis (SGS(Sentence), targetWord(ID,Lemma,PoS))

Input: SGS[value.0,value.1], targetWord[ID,LEMMA,POS]
Parameters: windowSize, weightingFunction

```
1: linearDistance = 1
2: targetWordMass = setMass(targetWord[POS])
3: for edge in SGS do
4:    if edge[value.0] is equal to targetWord[ID] then
5:       distances[edge[value.1]] += linearDistance
6:       add edge[value.1] to propagation
7:       remove targetWord[ID] from propagation
8:       windowSize = windowSize-1
9:    end if
10: end for
11: while windowSize > 0 do
12:    linearDistance =+ 1
13:    for id in propagation do
14:       for edge in SGS do
15:          if edge[value.0] is equal to id then
16:             distances[edge[value.1]] += linearDistance
17:             add edge[value.1] to propagation
18:             remove id from propagation
19:             windowSize = windowSize-1
20:          end if
21:       end for
22:    end for
23: end while
24: coOccurrenceValues = WeightingCoOcc(distances,weightingFunction)
```

Output: co-Occurrence Values of a Sentence (coOccurreceValues)

to continue the propagation. Then, according to the value of *windowSize*, it starts a new loop for all the IDs in the propagation list.

The algorithm assigns co-occurrence values after calculating the distances between the target word and the other words in the sentence (Algorithm 3). The Weighting Function may be of two types: a linear function that assigns a decreasing value to the words as the distance increases the target word or a GRAV[2] function based on the POS of the target word.

Finally, the algorithm actualizes the general matrix, adding the values generated for the target word for each context word and repeating the loop for the next target word.

When the corpus has been entirely processed, the algorithm converts the co-occurrence matrix into a correlation matrix. In the COALS algorithm, the Pearson Correlation is performed over the original matrix so as to generate the conditional rate instead of the raw rate of word-pair co-occurrence. The authors claim that computing *Pearson's correlation* between the occurrence of a word *a* and a word *b* can express the tendency of *b* to occur "more or less often in the vicinity of a than it does in general".

2 We will illustrate the GRAV function in section 4.2.2

Algorithm 3 WeightingCoOcc (SyntDistances(Sentence), weightingFunction)

Input: distances Map
Parameters: weightingFunction

```
for key in distances do
    contextWord = sentenceWord[key].value(LEMMA)
    if weightingFunction is Linear then
        coOccurrenceValues[contextWord] = (distances[key]*-1)+(WindowSize+1)
    else if weightingFunction is GRAV then
        coOccurrenceValues[contextWord] = Mass^2/distances[key]
    end if
    if coOccurrenceValues[contextWord]<0 then
        coOccurrenceValues[contextWord] = 0
    end if
end for
```

This normalization converts the co-occurrence values into values that range between -1 and 1. Converting all resulting negative correlations, which represent anti-correlated words, to 0, the matrix becomes more sparse and the model's performance may improve. Rohde, Gonnerman, and Plaut (2006) compared the COALS algorithm with a similar algorithm like HAL, which differs from the former mainly in this feature, obtaining considerably better results.

Taking into account the sentence presented in table 2, a word window of 5 and a Linear weighting function, we obtained the matrix shown in table 3.

Table 3
The Matrix generated by the presented model of sentence in table 2

	Kennedy	served	as	president	of	the	United	States	until	his	assassination
Kennedy	0	5	3	4	2	2	2	3	3	3	4
served	5	0	4	5	3	3	3	4	4	4	5
as	3	4	0	5	3	3	3	4	2	2	3
president	4	5	5	0	4	4	4	5	3	3	4
of	2	3	3	4	0	4	4	5	1	1	2
the	2	3	3	4	4	0	4	5	1	1	2
United	2	3	3	4	4	4	0	5	1	1	2
States	3	4	4	5	5	5	5	0	2	2	3
until	3	4	2	3	1	1	1	2	0	4	5
his	3	4	2	3	1	1	1	2	4	0	5
assassination	4	5	3	4	2	2	2	3	5	5	0

The matrix shown in table 3 is dense. Matrix density is particularly pertinent to short sentences, but the algorithm generally produces denser matrices with high values of word-window because, while words in syntactic structures are much more interconnected, a value higher than 5 tends to propagate throughout the sentence. This is an obvious consequence of using syntactic parsing data in matrix construction (Sahlgren 2008).

Table 4 shows the results of applying the Pearson Correlation to the Matrix presented in Table 3. Matrix density decreases markedly in Table 4. The matrix becomes even more sparse when lower values of word-window are used.

Table 4
The matrix after Pearson Correlation

	Kennedy	served	as	president	of	the	United	States	until	his	assassination
Kennedy	0	0,219	0,086	0,116	0	0	0	0	0,168	0,168	0,179
serve	0,219	0	0,112	0,105	0	0	0	0	0,192	0,192	0,179
as	0,086	0,112	0	0,201	0,119	0,119	0,119	0,136	0	0	0
president	0,116	0,105	0,201	0	0,146	0,146	0,146	0,133	0,014	0,014	0
of	0	0	0,119	0,146	0	0,248	0,248	0,252	0	0	0
the	0	0	0,119	0,146	0,248	0	0,248	0,252	0	0	0
united	0	0	0,119	0,146	0,248	0,248	0	0,252	0	0	0
states	0	0	0,136	0,133	0,252	0,252	0,252	0	0	0	0
until	0,168	0,192	0	0,014	0	0	0	0	0	0,295	0,298
his	0,168	0,192	0	0,014	0	0	0	0	0,295	0	0,298
assassination	0,179	0,179	0	0	0	0	0	0	0,298	0,298	0

4.2 SD-W2 parameter selection

In a preliminary experimentation phase, we tested different criteria for the parameter selection of the presented algorithm. These parameters are:

- Syntactic Distance
- Weighting function

In addition, we tested the Singular Value Decomposition (SVD) algorithm (Rohde 2002) in order to vary the dimensionality of the final matrix. SVD is a method for the linear decomposition of a matrix into independent components adopted for the first time by Landauer and Dumais (1997) in Distributional Semantics for Latent Semantic Analysis. The LSA model uses the SVD algorithm to produce a better simulation of human word-learning. The authors claim that SVD *embodies the kind of inductive mechanisms that they want to explore and provides a convenient way to vary dimensionality*. Since SVD did not greatly change performance in our preliminary test, we decided not to add a dimension reduction algorithm to our model.

4.2.1 Syntactic Distance

The selection of window size in word-based distributional semantics models can consider a neighborhood ranging from one to 1000 words (Sahlgren 2008). Schutze (1992a) proposes a window size of 1000-1200 words, claiming that word size is more important than the number of words taken into account in context construction. Yarowsky (1992) and Gale, Church, and Yarowsky (1995) use 100-word windows. Lund and Burgess (1996) use 10-word windows in HAL and Rohde, Gonnerman, and Plaut (2006) in COALS, suggest using 4-word windows. Although there are no word-windows in Syntactic methods, they extract co-occurring words from a dependency graph by defining a list of paths. The length of this paths plays the same role as the dimension of the word-window in linear models.

With this work we present a syntactic model in which we replace specific dependency paths with a generic syntactic window in which all the words related with a target word within a variable syntactic distance are included in its context. The value of Syntactic Distance, in this way, work exactly as a variable word-window, with the difference that it was unclear how many words the model would include in the context.

For example, if we can find more than one word in a sentence at a distance of 1, the number of words taken into account grows when this distance value increases.

The distance value used in our experiment ranges from 1 to 5.

4.2.2 Weighting Functions

We tested the system using a *linear weighting function* in which the co-occurrence value ranges from the dimension of the word-window to zero, decreasing once the syntactic distance grows. With $d = SyntacticDistance$ and $w = window - size$ the co-occurrence value c is calculated as:

$$c = (-d + (w + 1))$$

In the sentence *Kennedy served as a president of the United States until his assassination*, taking into account the word *Kennedy* as Target Word and a window-size of 2, the algorithm assign $-1 + (2 + 1) = 2$ to syntactically adjacent words (*served*), $-2 + (2 + 1) = 1$ to words at distance 2 (*president* and *assassination*), $-3 + (2 + 1) = 0$ to word at distance 3, and so on, setting all the negative values to zero.

In order to improve the variability of co-occurrence values, we also tested a different function, related to the words' syntactic features and using the parser graph. We were inspired by the idea that some words with certain POS tags (i.e. function words) tend to be very frequent and do not convey semantic information (Rohde, Gonnerman, and Plaut 2006). In COALS, these words were excluded from the final matrix. Our aim is to preserve this information but introduce proportional weights for each POS.

The parsed sentences are graphs in which words are nodes and relations are directed edges. By considering POS tags as nodes, we extract the total of the relationships in which each POS tag is involved in a section of one million words of the British National Corpus (BNC), parsed with the Stanford Core-NLP Parser Package.

Since we convert dependency graphs into undirected graphs (we take into account relations both pointing towards a node and starting from the node), we choose to use the total percentage of relations (in+out) as the *Mass* of a word. In our opinion, this value reflects the centrality of the POS tag in the sum of sentence networks of the corpus and proposes a set of values with greater significance and variability.

The main idea is to give each word a weight based on its influence on the syntactic graphs. Nouns and Verbs, for example, have a high *Mass* value that reflects their centrality in the structure of the sentences.

Definition 2

The **Mass** of a word is equivalent to the ratio of the number of incoming and outcoming arcs of a given POS and the total number of relations in a 1 million word Corpus extracted from the BNC.

For example, Nouns are involved in 41% of the relations in the first one million words of the BNC. This means that out of 100 arcs in the sum of the dependency graphs, 55 point to and 27 start from a Noun. If we observe the dependency graphs, we will see that Nouns are pointed to by Determiners (*the book*), Adjectives (*beautiful girl*) and other Nouns (*city center*). Conversely, they point mainly to Verbs, Nouns and Prepositions. If we take Determiners or Adjectives into account, these are involved in 5% and 7% of edges and, in the vast majority of cases, they point only to Nouns.

When we score the co-occurrence of the terms included in our matrix, we give higher values to categories that we consider central to our semantic analysis, without

completely eliminating categories that include non-content words. In the final matrix, this difference only affects rows, because the score is influenced only by the mass of the target word. In addition, we square the values so as to increase the difference between the POS tags and to obtain better results.

The influence of the *Mass* of a word must decrease as the distance increases, so the weight function, called GRAV, is calculated using the following formula:

$$GRAV = Mass_t^2 / Distance_{t,w}$$

$Mass_t$ indicates the weight of the POS tag of the Target Word and $Distance_{t,w}$ is the syntactic distance between the target word and the co-occurring word. In this perspective, each word may be considered an object with a certain *syntactic Mass* and produces an *attraction* over its neighbor words that is stronger if the POS of the word tends to be central in sentence networks. The attraction decreases as the distance increases.

In the sentence *Kennedy served as a president of the United States until his assassination*, taking into account the word *Kennedy* as Target Word and a window-size of 3, the algorithm assign $41,26^2/1 = 1.702,3876$ to syntactically adjacent words (*served*), $41,26^2/2 = 851,1938$ to words at distance 2 (*president* and *assassination*), $41,26^2/3 = 567,4625$ to word at distance 3. Conversely, the word *the* will obtain a co-occurrence value of $5,32^2/1 = 28,5156$ with its adjacent word *United*, $5,32^2/2 = 14,2578$ with words at distance 2 and $5,32^2/3 = 9,5052$ with words at distance 3.

4.3 Best Configuration

The algorithm presented in the previous section was developed in Java, using the *sspace* package developed at the Natural Language Processing group at UCLA[3]. The package contains algorithms and tools for constructing a distributional model and a set of compiled well-known classic algorithms such as LSA, HAL, DVS, and COALS.

In order to test the parameter of the model, we use the British National Corpus (Leech 1992), a 100 million-word Corpus of English, including written and spoken language. The corpus was parsed with the Stanford Core-NLP Parser Package (Manning et al. 2014).

The dictionary we used as Base-Map includes more than 18,000 words with high-frequency values extracted from the BNC[4] (more than 400 occurrences in BNC), which correspond to 12,024 lemmas.

With a view to testing our model, we defined an *optimal model* with a parameter setting that maximizes the experimental results. To test the parameter selection, we used the Rubenstein and Goodenough similarity test (Rubenstein and Goodenough 1965), as suggested by Padó and Lapata (2007). The original test calculated the correlation between the evaluations of semantic similarity performed by groups of humans on two lists of 24 theme words. The experiment involved 65 noun pairs scored on a 0-4 scale. The original model calculated a Pearson correlation (Pearson's *r*) coefficient of 0.85 when applied to similarity ratings between annotators.

We obtained the best results with no matrix reduction applied. The differences between weighting functions and syntactic distance (D) are shown in table 5.

3 The sspace package is freely downloadable at https://github.com/fozziethebeat/S-Space/wiki
4 Frequency list download at http://www.kilgarriff.co.uk/bnc-readme.html

Table 5
Evaluation of different parameters application on Rubenstein and Goodenough test

Syntactic distance	Linear WF	GRAV WF
1	0.65	0.64
2	0.656	**0.661**
3	0.63	0.64
4	0.61	0.63
5	0.59	0.62

The results presented in table 5 show a minimal variation between the application of the two weighting functions, with a slight advantage for the GRAV function. Conversely, the syntactic distance shows bigger variations with a clear propensity for models with the syntactic distance set as 2. The selected parameters were:

- words and Dimensions: 12,024
- Distance: 2
- Weighting function: GRAV

Once the parameters producing the best results are established, we also train the model on a larger corpus, the *WaCkypedia English corpus* (Baroni et al. 2009), a 2009 dump of English Wikipedia, cleaned and parsed with MaltParser (Nivre, Hall, and Nilsson 2006), of about 800 million tokens.

5. Experiment

This section presents a series of experiments on which the methodology described in section 4 was tested. As announced in section 1, our results on three tasks will be compared with other word-window models. Since we found an optimal configuration for our parameter, we retrain the model using a larger corpus.

The experiments we report in the paper are related to the classic semantic tasks addressed by many authors in DS literature:

- Semantic Similarity: a set of experiments in which the algorithm must express a similarity value between two words in a list of pairs already classified by humans. The correlation between the values given by the model and the human's values represents the algorithm's assessment score.
- Synonymy: this kind of text is based on synonymy tests generally proposed to foreign students of English during their assessment. The test consists of choosing the correct synonym for a word from four alternatives.
- Single-Word Priming: the test consists of finding the strongest association between a set of words representing six different lexical relations (synonymy, antonymy, super-subordination, category coordination, conceptual association, and phrasal association).

- In addition to these experiments, we will introduce a new task related to the concept of *selection* as conceived by Harris. This measures the similarity of a group of nouns belonging to a specific class with a verb that selects that class as the subject or the object.

In order to gain a clearer idea of the obtained results, we compared the two trained models (WaCkypedia and BNC) with other state-of-the-art models:

- Contextualized Models such as BERT (Devlin et al. 2019) or ELMo (Peters et al. 2018), as reported in Lenci et al. (2022) and Wang, Cui, and Zhang (2021);
- the results of similar models such as COALS and DVS, as reported in its original papers and by Jurgens and Stevens (2010);
- the results of classic models such as LSA and HAL, as reported by various sources;
- the results of COALS and Word2Vec (Mikolov et al. 2013b, 2013a) trained on the BNC corpus.

The data set and the experiment on the *argument selection task* will be presented in section 5.4; section 5.1 shows the results of our model on Semantic Similarity Task, in 5.2 we present the experiments on synonymy tasks and in 5.3 we replicate the semantic priming experiment presented in Padó and Lapata (2007) using our model.

5.1 Semantic Similarity Task

Semantic Relatedness is an important research topic in NLP (Taieb, Zesch, and Aouicha 2020). To verify the effectiveness of semantic relatedness extraction methods, the computational results are usually compared with human judgments. The cost of manual annotation of relatedness values limits the size of this kind of evaluation data set. Besides, a careful selection of the words is required.

We decided to test our algorithm on four Semantic Similarity data sets that have been used as a test set by many other authors. In particular, we tested our optimal model on the following data sets:

- **Rubenstein and Goodenough similarity pairs** (Rubenstein and Goodenough 1965) (RG65): this data set, described in section 4.3, is one of the most frequently used in evaluating DS models on semantic similarity. We compared our results with the results reported by Padó and Lapata (2007); Rohde, Gonnerman, and Plaut (2006); Landauer and Dumais (1997); Lund and Burgess (1996) and compared the evaluation of the same models trained on different corpora presented by Jurgens and Stevens (2010). In accordance with Rohde, Gonnerman, and Plaut (2006), we also tested the model on a reduced RG data set of 52 pairs of words, produced by deleting 5 ambiguous words.
- **Miller and Charles ratings** (Miller and Charles 1991) (MC30): this is another common similarity data set, which includes 30-word pairs of the RG65 manually evaluated by 38 subjects. The words selected for the MC30 data set have higher frequencies than the original RG set. For this subset,

we used both the original one and a reduced version with 5 ambiguous words deleted and 24 pairs.

- **WordSimilarity-353 Test Collection** (Finkelstein et al. 2001) (WS353): this data set includes 353 pairs rated by 13 or 16 subjects on a 0-10 scale. The set includes the MC30 pairs, proper names (such as *Arafat* or *Maradona*), word associates that are not synonymous (*tennis-racket*), adjectives, or gerunds. The words of WS353 are, in general, more common than those in RG.
- **SimLex999** (Hill, Reichart, and Korhonen 2015) (SL999): SimLex-999 is a gold standard resource for semantic similarity tasks. Five hundred native English speakers produced the resource: it contains 999 adjective, verb, and noun concept pairs. The experiment was designed as shown in Hill, Reichart, and Korhonen (2015), in order to compare the optimal model with the performance presented in that paper on the whole set and *abstract-concrete* subset and *Adjective-Noun-Verb* subset.

Table 6
Comparison of different algorithms on different Semantic Similarity Data Sets

Algorithm	Corpus	RG65	MC30	WS353	SimLex999
SD-W2	BNC	0.682	0.605	0.527	0.303
COALS	BNC	0.569	0.453	0.427	0.22
DVS	BNC	0.62	-	-	-
W2V (CBOW)	BNC	0.678	0.647	0.566	0.324
SD-W2	Wikipedia	**0.842**	**0.76**	0.614	0.394
BERT.L4	BookCorpus and Wikipedia	0.81	-	0.62	**0.55**
BERT.avg	Wikipedia	0.812	-	0.594	0.468
ELMo.avg	Wikipedia	0.668	-	0.583	0.436
SG	Wikipedia	0.752	-	0.610	0.394
CBOW	Wikipedia	0.727	-	**0.627**	0.380
LSA	Wikipedia	0.681	-	0.614	-
HAL	Wikipedia	0.261	-	0.195	-
COALS	USENET	0.682	0.671	0.626	-
HAL	USENET	0.153	0.319	0.311	-
LSA	USENET	0.656	0.731	0.599	-

In table 6, we present our results on the 4 Word Similarity tests included in the experiment. We organized the table in three section on the base of the corpus used to train the model.

The results of other algorithms were taken from Rohde, Gonnerman, and Plaut (2006) for the models trained on the USENET Corpus (1.2 billion words); from Jurgens and Stevens (2010) for LSA and HAL trained on WIKI corpora (respectively 600 and 900 million words); and from Padó and Lapata (2007) for the DVS model.

The scores for Contextualized Models were collected from two sources: Lenci et al. (2022) analyzes three different types of BERT embeddings: BERT.F4 which uses the sum of the embeddings from the first four layers; BERT.L4 which uses the sum of the embeddings from the last four layers; and BERT.L which uses the embeddings from the last layer. In all cases, the authors used the bert-large-uncased model (pretrained on *BooKCorpus*[5] and English Wikipedia). We report only the model which records the best

5 BookCorpus is a corpus of 11.038 unpublished books

scores (BERT.L4).
Wang, Cui, and Zhang (2021) adopts three different methods to use static similarities from BERT and ELMO, but we selected the one which obtained the best results (defined as BERT.avg and ELMo.avg by the authors). From the same paper, we also report the score of Skip-Gram (SG) and CBOW. All the models presented in Wang, Cui, and Zhang (2021) are trained on a Wikipedia Dump (1.1 billion tokens).

For COALS-BNC we used the *sspace package* and set the same parameters specified by the authors, 14,000 dimensions for each vector, 15,000-word vectors, and a list of *syntactic words* and punctuation excluded from the calculation of the matrix. For W2V-BNC we used the Python Gensim package[6], which uses CBOW as the default model, with automatic frequent phrases detection, a window-dimension of 5 and 200 dimensions.

In accordance with Rohde, Gonnerman, and Plaut (2006), we used the rank-order Correlation (Spearman's rho) to calculate the correlation between our results and human ratings, and we used the best-fit exponential scaling of similarity scores: scores of less than 0 are set to 0, and positive scores are replaced by $S(a,b)^t$ where $S(a,b)$ is the similarity score obtained, and t is an exponential that maximizes the model's correlation. A value of $t > 1$ increases sensitivity at the high end of the rating scale and $t < 1$ at the low end. We used a $t = 0.7$ for the SD-W2 model and W2V and 0.6 for COALS trained on BNC. The similarity values have been generated using Pearson's correlation for SD-W2-BNC and Cosine Similarity for the other models (including SD-W2-Wiki).

Concerning *SimLex-999*, we also followed the experiment conducted by Hill, Reichart, and Korhonen (2015) who tested their data set on a representative set of DS models such as LSA, VSM (Kiela and Clark 2014) or Word2Vec (Mikolov et al. 2013a). In table 7 we compare the correlation of both SD-W2 models with the correlation of LSA and W2V trained on the RCV1 Corpus (~ 150 million words) (Lewis et al. 2004) with two different window sizes (10 and 2) as reported by Hill, Reichart, and Korhonen (2015), and with COALS and Word2Vec trained on BNC.

Table 7
Comparison of SD-W2, COALS-BNC, W2V, and LSA on SimLex-999

Algorithm	SimLex-999	Most Associated 333	Adjectives (111)	Nouns (666)	Verbs (222)	Concrete (250)	Abstract (250)
SD-W2-Wiki	0.394	0.212	0.421	0.455	0.191	0.425	0.296
W2V-Wiki	0.414	0.260	-	-	-	-	-
SD-W2-BNC	0.303	0.107	0.413	0.359	0.08	0.315	0.227
COALS-BNC	0.220	0.017	0.338	0.253	0.034	0.212	0.200
W2V-BNC	0.324	0.057	0.463	0.342	0.170	0.339	0.369
LSA-RCV1 (2)	0.233	0.009	0.375	0.270	0.085	0.226	0.185
LSA-RCV1 (10)	0.238	0.070	0.272	0.298	0.008	0.325	0.209
W2V-RCV1 (2)	0.282	0.178	0.436	0.303	0.161	0.248	0.306
W2V-RCV1 (10)	0.266	0.176	0.406	0.278	0.114	0.236	0.309

Table 7 refers to different subsets of SimLex-999. The correlation for the whole set is shown in the second column. The third column reports the value of a subset of 333 most strongly associated concepts, according to the University of South Florida Free Association Database (USF) (Nelson, McEvoy, and Schreiber 2004). Association data were generated by human subjects who produced a set of associated words for 5000 concepts.

6 https://radimrehurek.com/gensim/models/word2vec.html

5.1.1 Discussion

In table 6 we present the results of our model compared with many state-of-the-art models, in relation to corpora of different kinds and dimensions. This comparison allows us to study the importance of the corpus dimension and typology on the generation of co-occurrence values. Starting from the models trained on BNC, we must underline that the CBOW model of Word2Vec reaches higher results compared with SD-W2, for all four data-sets. As pointed out also by Hill, Reichart, and Korhonen (2015), SimLex-999 is *notably more challenging* than the other data-sets: nevertheless, the results of W2V trained on BNC also surpass the scores of the same model trained on RCV1 as reported by Hill, Reichart, and Korhonen (2015) and presented in table 7. Concerning the other data-sets, SD-W2 achieves better results than COALS and DVS, from which it draws inspiration and obtains similar results to W2V.

If the corpus dimension is increased, the results of our model become comparable to those of the contextualized models. Regarding the smaller data-sets, SD-W2 shows the best results with a precision of 0.842, overcoming both BERT (0.81) and ELMo (0.69), but also the two Mikolov models Skip-Gram and CBOW (respectively 0.75 and 0.73). With bigger data-sets such as Word-Sim353 and Sim-Lex999 the performance of SD-W2 decreases, but they are still comparable with the results of other models trained on a Wikipedia Corpus. In fact, for WS353, our results are in line with those of LSA, SG and ELMo and slightly lower than those of CBOW and BERT.L4. For SimLex999, the results of SD-W2 are similar to SG and CBOW but significantly lower than BERT and ELMo.

We tested SD-W2 also on the subsets of SimLex999 and compared the results with those presented by Hill, Reichart, and Korhonen (2015) and with the models trained on the BNC. In table 7, we present the results of SD-W2 compared with W2V, both trained on Wikipedia, but also the results of the same models trained on BNC. We also compared our model trained on BNC with LSA and W2V trained on RCV1 (similar in size to BNC).

The performance of our model varies according to subset and training corpus: if we consider the models trained on Wikipedia, we can compare SD-W2 only with W2V and only for the full data-set and the Most Associated 333 pairs. In this case, the results are very similar, especially with the full data-set. Regarding the models trained on the smaller corpora, if we consider the models presented in Hill, Reichart, and Korhonen (2015), we obtain high results over the whole Simlex, the Most Associated 333, and the subset of Nouns. We performed worst over the other subsets such as Verbs and Abstract Nouns.

5.2 Synonym Detection

Landauer and Dumais (1997) tested LSA on the *Test of English as a Foreign Language* (TOEFL) for the first time. In the paper, the TOEFL test was reduced to 80 questions (items) requiring the synonym of a given target word to be identified in a group of 4 words. The original test also provided a small clause context to the target word that Landauer had deleted in his computational experiment. After this test, many other tests have been used to evaluate DS models, such as the ESL (*English as a Second Language*) (Turney 2001) test or the *Reader's Digest Word Power test* (Jarmasz and Szpakowicz 2004). In particular, the ESL test consists of 50 items that tend to include words with higher frequencies than the TOEFL items. ESL items are based on a more subtle discrimination of meaning. For the target word *passage*, for example, the four alternatives are *hallway*, *ticket*, *entrance*, *room* and the solution is the word *hallway*.

In this paper, we will test our model on the TOEFL and ESL tests. The results are shown in Table 8[7].

Table 8
Comparison of different algorithms on TOEFL and ESL tests

Algorithm	Corpus	TOEFL	ESL
SD-W2	BNC	0.69	0.49
COALS	BNC	0.75	0.46
DVS	BNC	0.73	-
W2V	BNC	0.75	0.64
SD-W2	WIKI	0.76	0.54
BERT.L4	WIKI	0.89	0.60
HAL	WIKI	0.50	0.31
LSA	WIKI	0.61	0.54
COALS	USENET	0.86	0.52
HAL	USENET	0.56	0.26
LSA	USENET	0.53	0.43

In these experiments, we calculated the semantic similarity between the target word and each item's words. We took the word with the highest similarity value as the correct answer and then calculated the accuracy by counting the correct answers.

Considering a human average score of 64.5% for the TOEFL test, we can affirm that SS-W2 surpassed the human rating.

5.2.1 Discussion

The semantic similarity task tackled in this section includes two classic experiments: TOEFL and ESL. In comparing different models, the use of the same (training) corpus would have guaranteed consistent, better aligned results (Padó and Lapata 2007). Nevertheless, it would have been a major process to train a different model on BNC, so we must rely on the accuracy values reported in other papers. Table 11 shows the accuracy of the same DS models presented in the previous section, so we inserted only two scores achieved by models trained on the BNC corpus. Regarding the DVS model, we only have information on the TOEFL test because it is the only test the authors considered in their experiment.

According to the accuracy highlighted by Padó and Lapata (2007), we know that the PMI-IR model (Turney 2001) trained on BNC attains 61.3% accuracy, while the original model trained on a large Web-based corpus achieves 72.5%.

As for the similarity task, we report the results of our model trained on the two different corpora, BNC and Wikipedia. The results of our model are below expectations both for the one trained on BNC and for the one trained on Wikipedia and for both data-sets. If we do not consider the older models, SD-W2 obtains very low results for the two data-sets, reaching the best precision of 0.759 on TOEFL when trained on Wikipedia which is similar to the precision of the other model trained on a smaller

7 A complete list of TOEFL results for DS models is shown on
`https://aclweb.org/aclwiki/TOEFL\_Synonym\_Questions\_(State\_of\_the\_art)`

corpus. Although our model achieves the average precision score for count models (Baroni, Dinu, and Kruszewski 2014) on the TOEFL, we believe its precision to be too low when compared to the output of other models like COALS.

In conclusion, the results of SD-W2 are above the average of the tested models both for TOEFL and ESL. In Lapesa and Evert (2014) the authors claim that the parameters affecting the accuracy of the model for the TOEFL test are the distance metric, the score and the transformation. Cosine similarity, for example, produces better results than other metrics, while the association measures based on significance tests achieve the best results. Window-size might also affect model performance, the best results being achieved with a window-size of 2. Nevertheless, Lapesa and Evert (2017) tested the best parameters for dependency-based DSM, and the authors found that the parameters with a strong impact are metric, score and transformation. Analyzing the results of Lapesa and Evert (2017), we can impute the lack of precision of SD-W2 mainly to the absence of dimension reduction.
Also Bullinaria and Levy (2007) analyses the importance of different parameters on many semantic tasks. For the TOEFL task, for example, the models tested in the paper obtain the best results with small window-size. Since in our model the windows size do not correspond to a specific number of terms, we can't really control the number of words that belongs to the context of a given term and this can negatively affect the precision of SD-W2. Nevertheless, the conclusions of Bullinaria and Levy (2007) contrast with those of Lapesa and Evert (2014) regarding the dimension reduction.

5.3 Single-Word Priming

Inspired by Padó and Lapata (2007), we decided to also test the SD-W2 model on a simulation of semantic priming. This task is addressed in other studies (Lund and Burgess 1996; McDonald and Brew 2004) and entails the exposure of semantic similarity or dissimilarity between words. According to Padó and Lapata (2007, p. 180), "if dependency-based models indeed represent more linguistic knowledge, they should be able to model semantic priming better than traditional word-based models".

The experiment is based on the Hodgson (1991) single-word priming study. The underlying principle is that the presentation of a *prime* word like *clown* could facilitate the lexical decision on a *target* word like *circus*. Hodgson proposed an experiment in which the human subjects must take a decision about 144 pairs of words belonging to six different lexical relations: synonymy (*trash-garbage*), superordination or subordination (*fuel-gas*), category coordination (*rectangle-circle*), antonymy (*enter-exit*), conceptual association (*clown-circus*), and phrasal association (*foreign-language*). The goal of the experiment was to investigate the influence of each lexical relation on the prime effect. The paired words were selected from different POS (Nouns, Verbs, and Adjectives) and represented an unambiguous example of the relation type. The results of the original experiment demonstrate that there is an equivalent priming effect for the six lexical relations.

This experiment was used in McDonald and Brew (2004) to test the ICE (*Incremental Construction of Semantic Expectations*) model. In Padó and Lapata (2007) the 143 original pairs (one synonymy pair was lost) were reduced by deleting pairs with at least one low-frequency word. The authors set the Lexical Relation and prime (related, unrelated) as independent variables. The dependent variable representing the quantity being measured is the semantic distance between the prime and the target. The distance between Related and Unrelated prime-target pairs simulates the priming effect. Since the Unrelated primes were not provided in the description of the original experiment,

both DVS and ICE models used the averaged distance of a target to all other primes of the same relation as unrelated primes.

In order to measure the prime effect and compare the results with the DVS model, we performed a two-way analysis of variance (ANOVA) on the data generated by SD-W2, COALS-BNC and W2V-BNC. Lexical Relation (six levels) and prime (two levels) were the factors. SD-W2 showed a strong prime effect as with BNC (F(1,135) = 257.64, MSE = 2.15, $p < 0.01$) as with Wackypedia (F(1,135) = 435.64, MSE = 3.52, $p < 0.01$). The value of p is significant (< 0.01) and indicates a significant difference between Related and Unrelated pairs. Also COALS-BNC (F(1,135) = 163.92, MSE = 1.02, $p < 0.01$) and W2V-BNC (F(1,135) = 447.09, MSE = 10.05, $p < 0.01$) showed a significant prime effect.

Having determined that there are differences between Related and Unrelated prime targets, we need to quantify the magnitude of the prime effect. Padó and Lapata suggest using the Eta-squared (η^2) measure, often employed to calculate the strength of an experimental effect. The formula of Eta-squared is $\eta^2 = \frac{SS_{effect}}{SS_{total}}$, where SS_{effect} represents the variance (sum of square) created by one particular effect (the prime) and SS_{total} is the sum of the variance of all observations. It represents how the variability in the distance variable can be explained by priming (Related-Unrelated). DVS reports an eta^2 of 0.332. This means that DVS accounts for 33.2% of the variance. The eta^2 of SD-W2 trained on BNC is 0.477, while trained on Wackypedia is 0.566. COALS obtains 0.383. The eta^2 obtained by W2V-BNC is 0.613.

In order to verify the prime effect over all six relations, we produced different ANOVAs for each Lexical Relation. Table 9 reports the mean distance values for each relation in the Related and Unrelated condition. It also indicates the prime effect size for each relation for SD-W2, COALS-BNC, and DVS, calculated as Related-Unrelated.

Table 9
Mean distance values for the six Lexical Relations; Prime Effect size for SD-W2, COALS, DVS and W2V

Lexical Relation	Related	Unrelated	SD-W2 BNC Effect	SD-W2 WIKI Effect	COALS Effect	DVS Effect	W2V Effect
Synonymy	0.374391	0.141128	0.233262	0.294304	0.163129	0.165	0.514
Superordination	0.327209	0.126888	0.200321	0.287032	0.111652	0.106	0.386
Category coordination	0.340998	0.142409	0.198589	0.302349	0.124305	0.137	0.336
Antonymy	0.291833	0.142169	0.149664	0.197816	0.126387	0.165	0.409
Conceptual association	0.291064	0.122289	0.168775	0.172834	0.114011	0.083	0.404
Phrasal association	0.253054	0.125435	0.127619	0.132093	0.102564	0.043	0.282

5.3.1 Discussion

According to Padó and Lapata, the semantic priming must be modeled better by means of a model that can represent more linguistic knowledge. With this experiment, we point out that SD-W2 can show a reliable prime effect on the Hodgson experiment, surpassing the results of the other models tested on the same data set and trained with the same corpus. The significantly better results reached by Word2Vec reflect the advances of the DS models in the last years. The use of Neural Networks helps to produce better results although the corpus used was the same than other models.

Analyzing each Lexical Relation result presented in table 9, we observe a reliable prime effect on the six types for SD-W2. In particular, the model shows the best results with Synonymy and Superordination-subordination pairs (almost double the value obtained by COALS and DVS). Phrasal association, Conceptual association, and Category

coordination obtain decent results compared with the DVS model but similar to COALS. As for Antonymy, SD-W2 shows the worst prime effect.

Analyzing the similarity generated by single pairs, we notice that the Antonymy relation shows no critical issues but the closest Related-Unrelated values. In phrasal pairs, on the other hand, there is a general greater deviation between Related and Unrelated similarities, although in three cases the Unrelated value is higher than the Related one (*help-wanted*, *mountain-range*, and *pony-express*). While two of these values are very close, the value of the pair *pony-express* is considerably lower than the average distances of all the other primes. The low value obtained by Phrasal association pairs can be attributed to the nature of this association. In effect, it depends on *in-praesentia* relations and is strongly influenced by the co-occurrence of the two pair words in the corpus. For example, the words *pony* and *express* have high frequencies in BNC, but the sequence *pony express* only appears twice. Contrariwise, in Wikipedia, there are many pages in which the two words appears in association (movies, tv shows,sports and other categories).

5.4 Operator-Argument selection

In section 2 we stated that, according to Harris's distributional hypothesis, the context selection of DS models must include not the graphical context of a target word but its syntactic context since, according to Harris's theory, the distribution of a word must be associated with the relation between Operators and Arguments. This kind of relationship is a syntactic relationship and can be brought out by a dependency tree. This is why the SD-W2 model relies on syntactical dependency and selects all the words included within a syntactic distance range as contexts of the target word.
In order to test the ability of our model to detect Operator-Argument relations, we set up a new experiment in which the model must connect a class of nouns with the verb form that selects this class as a right or left argument. For the vast majority of verbs, subject or object selection includes very generic classes of nouns. The verb *to sleep*, for example, selects animate entities (*the dog*, *the child*, *John*, etc.) as likely subjects, like many other verbs. A transitive verb such as to listen presents a similar distribution to sleep for the subject and a huge selection of nouns as the object.

For the *Operator-Argument selection test*, we needed a set of verbs whose distribution must be restrictive. A verb like *to smoke*, for example, includes the very restricted class of "smokable items" as the object. The word *cigarette* can be selected as the argument in a wide range of verbs with variable likelihood. Whereas, if we consider the information that the Operator and the Argument mutually exchange, we must find a stronger similarity between the noun and the verb *to smoke*. Following this hypothesis, we built a data set of verbs with restricted arguments.

This data set is based on the syntactic classes of verbs collected by the *Lexicon-Grammar Theory* (LG). LG, which is deeply connected to the Operators-Arguments theory, determines the structure of a large number of verbs (Gross 1975) that were classified on the basis of their shared syntactic features. Since there are only specific LG tables of English verbs (mainly phrasal verbs), we relied on the Italian classification (Elia 1984; Vietri 2004) from which we selected two classes of verbs with restricted arguments: class 2B and class 20R.

Thanks to this classification, we were able to extract, for example, all the intransitive verbs with one restricted argument (*to bark*, *to derail*, *to erupt*, etc.) from class 2B, or transitive verbs with restricted objects (*to smoke*, *to drink*, *to celebrate*) from class 20R.

Class 20R includes 77 verbal uses characterized by a syntactic structure of the kind $N_0VN_{1restricted}$. The verbs of class 20R present only one complement (direct object) which is strongly restricted to one or a specific class of objects. We select 25 verbs from this class which present a very restricted selection and are not ambiguous or used metaphorically.
Class 2B includes 45 intransitive verbs with a structure $N_{0restricted}V$. As for class 20R, the subjects present a selection of nouns restricted to one specific class. Likewise in this class, many verbs used metaphorically have been discarded.

Hence, 70 Italian verbs were selected. These verbs were then translated and the 68 which keep the same properties in both languages were selected. From the list of 68 English verbs, we selected a restricted group by deleting verbs that feature a restricted argument only in one interpretation (*to quote*, *to cultivate*), with very low frequencies (*to erupt*, *to engrave*, *to rebind*), and with a metaphorical use (*to roar*, *to shine*)

The final list included 26 verbs that were used to generate sets of 4 nouns, which can figure as the restricted subject or the restricted object of these verbs. The groups include nouns that must represent both prototypes of the class of nouns required by the verb and more peripheral nouns, with the least possible ambiguity. The nouns of one group may occasionally appear again in another group.

We decided to include some verbs with a very similar distribution, such as *cook* and *fry*, and test the models with subtler differences.

The problem of choosing a group of nouns that work as the subjects or objects of a given verb primarily applies to verbs with similar meanings. When we selected the group of nouns for the verb *to wear*, we freely selected nouns from the list of clothes. In fact, clothes represent the restricted distribution of objects for *to wear*. However, using a frequency criterion for representativeness, we look for the most representative and distinctive objects among the nouns of clothes (*shirt*, *hat*, *jeans* and *shoes*).

On the other hand, when choosing the nouns for *fry* or *cook*, which both select the same class of nouns (edible items or foods), we attempted to choose nouns that emphasise the variations between the two distributions. For *to fry* we selected *potatoes*, *chips*, *eggs* or *bacon*. Since *fry* can be considered as a subclass of *cook*, the latter can also select all those elements, but with less probability than *bean*, *pasta*, *rice* and *bread*.

The groups of nouns were submitted to 40 human subjects to test their capacity to connect the arguments with the correct verb. The subjects were Italian undergraduates and master's degree or PhD students with good linguistic skills. They were asked to read the list of verbs and, for each group of nouns, choose a verb that can select all four nouns in the group as subject or object.

We calculate the precision as the number of correct answers (verbs correctly associated with the list of nouns they select) divided by the total number of questions (26).
The human subjects had issues with classes that can select very similar items such as *cook* and *fry* or *smell*, or *hunt*, *growl*, and *bark*, but in general, the average human precision is 0.923. This result validated the proposed group of nouns related to each verb: while most human subjects correctly associated nouns and verbs, some of them reported a precision range of 0.85 to 0.90. Only one subject scored 0.77. The fact that the human subjects confused *cook* with *smell*, which includes the nouns *flower* and *perfume*, or *hunt* with *bark*, which includes the noun *puppy*, indicates that many errors can be attributed to a cursory reading of the data.

Table 10 shows the selected verbs and the group of nouns.

Table 10
Data set for the Operator-Argument Selection Test

Verbs	Groups of selected nouns
fly	plane, robin, bird, helicopter
cook	bean, pasta, rice, bread
fry	potato, chips, egg, bacon
harvest	cereals, wheat, corn, grain
blossom	rose, violet, lily, daisy
growl	dog, monster, wolf, hound
gallop	rider, horse, pony, deer
asphalt	street, ground, square, road
boil	soup, water, milk, bean
hunt	fox, deer, elephant, bird
wear	shirt, hat, jeans, shoes
celebrate	marriage, wedding, festival, christmas
smoke	cigarette, cigar, tobacco, weed
drink	water, milk, whisky, juice
prune	pine, tree, oak, branch
prescribe	drug, medicine, pill, treatment
print	newspaper, book, picture, photo
drive	car, bus, train, truck
shear	hair, sheep, fur, goat
smell	garlic, cheese, flower, perfume
play	football, role, tennis, guitar
sing	song, carol, prayer, hymn
run	championship, race, marathon, tender
abort	baby, male, children, pregnancy
bark	dog, puppy, wolf, hound
bellow	bull, cow, elephant, ox

We tested the SD-W2 model with this data set by computing the best candidate verb for a group of nouns as the one with the highest average semantic distance from every noun. The precision of the SD-W2 model was 0.73 while COALS obtained 0.57. Word2Vec reaches a precision of 0.808.

Table 11
Comparison of different algorithms on Verb Selection Test

Algorithm	Corpus	Precision
SD-W2	BNC	0.73
COALS	BNC	0.57
W2V	BNC	0.81
SD-W2	WIKI	0.61
W2V	WIKI+GIGAWORD	0.65

As shown in table 11, we also tested SD-W2 model trained on Wackypedia and the GLOVE Word2Vec pre-trained model (6 billion of words from Wikipedia and Gigaword English Corpus) with 200 dimensions. Interestingly, those two versions, trained on larger corpora, obtain the worst results, underlining that the precision of the model, for this task, is not influenced by the corpus dimension, but by its content.

5.4.1 Discussion

With this experiment, we aimed to test the SD-W2 model's ability to detect the connection between a verb and the class of noun it selects as an argument. As for the Semantic Priming experiment, we think that dependency models must model this kind of relationship better because they explore the syntactic connection between words. Our experiment reveals two critical weaknesses: first, we compare our model only with COALS and Word2Vec; second, the data set is still incomplete and needs to be improved and tested by more human subjects.

In actual fact, we can only study the results of the SD-W2 model by exploring the critical issues we identified. The best model configuration (BNC) fails in the classification of seven groups: it confuses *fry* with *cook*, *asphalt* with *drive*, *prune* with *bark*, *shear* with *wear*, *abort* with *fly*, *bark* with *growl* and *bellow* with *fly*.

The model which reaches the best results was Word2Vec, which share some errors with SD-W2 (*prune*,*shear*,*abort* and *bellow*) but also confuses *run* with *gallop*.

In some cases, we expected the model to make the error, such as in the case of *to bark* and *to growl* which have a very similar meaning and select a similar group of items. The same goes for *to cook* and *to fry*.

As for *to prune* and *to bark*, we must attribute the error to the ambiguity of *bark*, which can also mean *the tough protective outer sheath of a tree trunk*. Since we train the model on a lemmatized corpus, we must use the dictionary form of the verbs, and we cannot disambiguate the meaning by using, for example, the past tense. This hypothesis is also confirmed by the error of Word2Vec.

The case of *to asphalt* and *to drive* is also clear, because for the latter verb what interferes may be a locative complement. In fact, drive has a higher similarity with *road* or *street*, much more than the similarity between the two words and *asphalt*.

With the verbs *abort* and *bellow*, the model confuses them with *fly*. In the first case, the word abort in BNC seems to be connected with the domain of computer science (as in he *terminates/aborts the program/process*) and it manifests a weak semantic association with all the words in the group. On the other hand, *fly* has higher similarity values with *baby* and *male* which are also related to the sphere of zoology. The word *male*, for example, has a strong association with the word *bird*.

The word *bellow* obtains similarity values with the four words in the group comparable to the ones obtained by *fly*, but the latter has a higher value with all the words. We observe the same behaviours in Word2Vec results for the two group of words.

In order to visualize the neighborhood of a verb like *to fly* or *to shear*, we developed a network composed of three levels of the verb's neighbors: we extracted the verb's 50 nearest objects (first-level objects) and their similarity scores, and performed a 10-object extraction (second-level objects) for each of the 50 first-level objects. We then replicated the same process for the second-level objects (third-level objects).

We generated a network in which the nodes are words and the weighted edges are similarity scores. We used Gephi (Bastian, Heymann, and Jacomy 2009) to build up the visualization and performed two specific graph algorithms. First, we calculated the degree of each node to point out words that frequently appear as the verb's nearest

neighbors; second, we ran the *Modularity Class* algorithm to calculate sub-communities of nodes and easily identify specific classes of words. Modularity Class (Blondel et al. 2008; Lambiotte, Delvenne, and Barahona 2008) was applied to the network with a resolution parameter of 2 to minimize the number of generated classes.

An example of the network is reported in figure 2, which shows two-word networks: the upper figure represents the neighborhood of the word *shear* (the yellow node). As can be seen, the words that emerge are all related to the domain of physics. The noun *shear* represents "a movement in the plates in the surface of the earth that causes them to change shape or break" and the verb *to shear* also refers to a deformation of a material substance in which parallel internal surfaces slide past one another.

The figure below refers to the word *fly* and shows the relation of the verb with its possible subjects. The Modularity Class identifies a class of *animals* (red nodes) in which the word *bird* stands out, but also a class of *vehicles* (green nodes), *places* (blue nodes), and *motion verbs* (black nodes).

The differences between a verb associated with the correct group of nouns (*fly*) and a verb where the system produces an error (*shear*) emerge clearly in this kind of visualization. In fact, in the network of *shear*, there is no sign of the nouns in the corresponding group. This is confirmed by figure 3 which contains the network of the word *celebrate*.

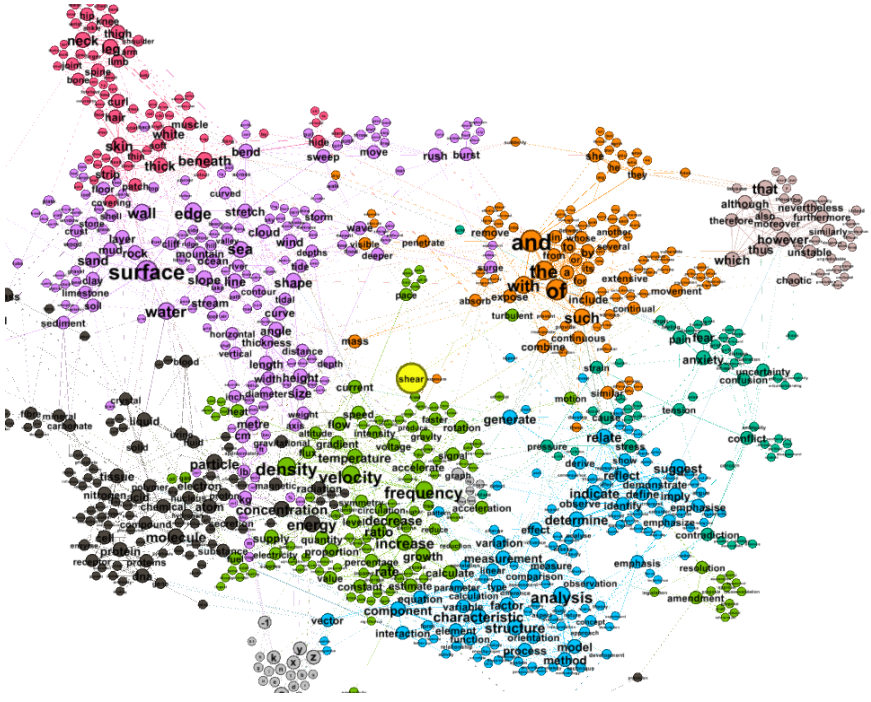
Among the neighbors of *celebrate* we find a group of words related to the temporal dimension (*weekend*, *day*, *evening*), music or arts in general (*concert*, *exhibition*), and events (*ceremony*, *festival*, *protest*).
This indicates that, in some cases, the problem may lie in the corpus where a specific meaning of a word is privileged and not in the model.

In general, SD-W2 obtained good results, compared to COALS-BNC. An analysis of the errors of our model points out that the words in the group associated with *to bark* and the words in the group associated with *to fry* belong to the same category of respectively Animals and Food, which are also selected by *to cook* and *to growl*. Even if a human subject can detect differences between these groups, we can consider this model's errors as minor. If these two groups of words are excluded from the data set or if the two automatic evaluations are considered exact, SD-W2 exceeds 80% accuracy, surpassing the score obtained by some of the human subjects who took part in the experiment.

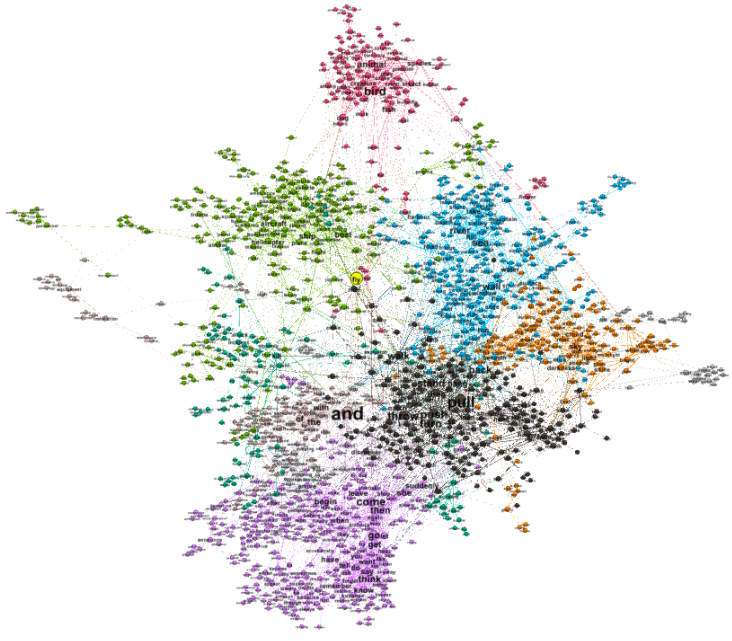
6. Conclusion

In this paper, we have presented a new model for Distributional Semantics. The model, called SD-W2, uses syntactic distances extracted from a parsed text to build a word's context. From the distributional hypothesis analysis conducted, we argue that the context of which Harris speaks is syntactic because every analysis of the meaning must be based on the Operator-Argument relation. To base our distributional analysis on the syntactic dependencies between words, we use a model that propagates the influence of a target word on its related words at a specific syntactic distance. In order to calculate this influence, we tested a linear method in which each word directly connected with the target obtains a higher value, and this value is decreased by 1 for more distant words. We also tested a different methodology in which we calculated the weight of the influence of the target word over the other words as a function of the percentage of the sum of its degrees divided by the distance.

Since we obtained the best results with the second methodology, we tested the model with the latter weight function in three experiments used by many other authors. The first family of experiments concerns semantic similarity. The model must replicate



(a) Shear



(b) Fly

Figure 2
Gephi network for two words (*shear* and *fly*); different colours correspond to different classes. Text size depends on the node degree.

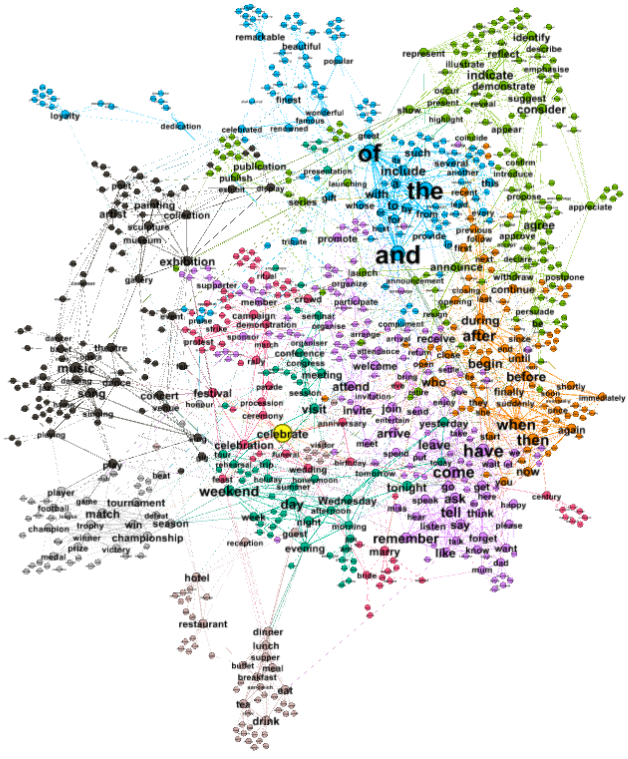


Figure 3
The network of *celebrate*

human judgment on the similarity between pairs of words. The performance of SD-W2 in this experiment exceeds the average of compared models and some cases are above the performance of the predict or contextualized models.

The second family of experiments deals with synonymy and is based on synonymy tests for foreigners. In this case, the results of SD-W2 are above the average of compared models. This result can be imputed to the lack of matrix dimensionality reduction, which is a parameter of crucial importance in this class of experiments. In the future, we plan to add an SVD reduction algorithm to the SD-W2 model to verify this assumption.

The third experiment regards single-word priming. The test is modeled over the Hodgson experiment (Hodgson 1991) and calculates the model's ability to simulate the prime effect on six different kinds of lexical relations. The results of SD-W2 are very encouraging for this task, surpassing those obtained for the compared models, except for Word2Vec which trained on BNC and obtained the best performance.

Finally, we subjected the model to a new experimental test regarding the operators' argument selection. Given a set of groups of four nouns belonging to a specific semantic class, the model must calculate the verb that selects the words of each group as an argument. The model reached a good accuracy score (73%) with errors that, in many cases, are due to the verb's ambiguity. Testing two models (SD-W2 and W2V) trained on different larger corpora, we realized that this parameter is not relevant to the task. The BNC corpus obtained the best results despite its reduced dimension. In the future

we plan to enlarge the experiment and test our model trained on different corpora in order to define the parameters that achieve the best results for the task.

We demonstrate that a dependency model could achieve good results without a large and expensive pre-processing phase. Comparing our model with a similar word-window model like COALS, trained on BNC, we demonstrate that SD-W2 can surpass COALS in almost all the selected tasks and with a comparable amount of pre-processing. Consequently, we demonstrate that growth in corpus size results in the exponential improvement in our model's performance. Training the model on a large corpus such as Wackypedia, its performance reaches the performance levels of DL-Based models in some cases.

References

Audet, Chad and Curt Burgess. 1999. Using a high-dimensional memory model to evaluate the properties of abstract and concrete words. In *Proceedings of the Twenty-first Annual Conference of the Cognitive Science Society*, pages 37–42, Vancouver, Canada, December. Citeseer.

Azzopardi, Leif, Mark Girolami, and Malcolm Crowe. 2005. Probabilistic hyperspace analogue to language. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 575–576, Salvador, Brazil, August. ACM.

Baroni, Marco, Raffaella Bernardi, and Roberto Zamparelli. 2014. Frege in space: A program for compositional distributional semantics. *Linguistic Issues in language technology*, 9(6):5–110.

Baroni, Marco, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Baroni, Marco, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247, Baltimore, United States, June.

Baroni, Marco and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Bastian, Mathieu, Sebastien Heymann, and Mathieu Jacomy. 2009. Gephi: an open source software for exploring and manipulating networks. In *Proceedings of the Third International AAAI Conference on Weblogs and Social Media*, volume 3, pages 361–362, San Jose, California, May.

Blondel, Vincent D., Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):1–12.

Boleda, Gemma. 2020. Distributional semantics and linguistic theory. *Annual Review of Linguistics*, 6:213–234.

Buchholz, Sabine and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the tenth conference on computational natural language learning (CoNLL-X)*, pages 149–164, New York City, United States, June. ACL.

Bullinaria, John A. and Joseph P. Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior research methods*, 39(3):510–526.

Burgess, Curt. 1998. From simple associations to the building blocks of language: Modeling meaning in memory with the HAL model. *Behavior Research Methods, Instruments, & Computers*, 30(2):188–198.

Burgess, Curt. 2001. Representing and resolving semantic ambiguity: A contribution from high-dimensional memory modeling. In *On the consequences of meaning selection: Perspectives on resolving lexical ambiguity*. American Psychological Association Washington, DC, pages 233–260.

Chersoni, Emmanuele, Enrico Santus, Philippe Blache, and Alessandro Lenci. 2017. Is structure necessary for modeling argument expectations in distributional semantics? In *IWCS 2017 - 12th International Conference on Computational Semantics - Long papers*, Montpellier, France, September.

Chersoni, Emmanuele, Enrico Santus, Chu-Ren Huang, and Alessandro Lenci. 2021. Decoding word embeddings with brain-based semantic features. *Computational Linguistics*, 47(3):663–698.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Elia, Annibale. 1984. *Le verbe italien: les complétives dans les phrases à un complément*. Schena; Nizet.

Finkelstein, Lev, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414, New York, United States, May.

Gale, William A., Kenneth W. Church, and David Yarowsky. 1995. Discrimination decisions for 100,000-dimensional spaces. *Annals of Operations Research*, 55(2):323–344.

Grefenstette, Gregory. 1992. Sextant: Exploring unexplored contexts for semantic extraction from syntactic analysis. In *30th Annual Meeting of the Association for Computational Linguistics*, pages 324–326, Newark, United States, June.

Grefenstette, Gregory. 1994. Corpus-derived first, second, and third-order word affinities. In *Proceedings of the 6th International Congress on Lexicography (EURALEX 1994)*, pages 279–290, Amsterdam, Netherlands, August. Rank Xerox Research Centre.

Gross, Maurice. 1975. *Méthodes en syntaxe: régime des constructions complétives*, volume 1365. Hermann Paris.

Harris, Zellig. 1968. *Mathematical structures of language*, volume 21. Interscience, New York, United States.

Harris, Zellig. 1976a. On a theory of language. *The Journal of Philosophy*, 73(10):253–276.

Harris, Zellig. 1976b. A theory of language structure. *American Philosophical Quarterly*, 13(4):237–255.

Harris, Zellig. 1988. *Language and information*. Columbia University Press, New York, United States.

Harris, Zellig. 1991. *Theory of language and information: a mathematical approach*. Oxford University Press, Oxford, UK.

Harris, Zellig S. 1946. From morpheme to utterance. *Language*, 22(3):161–183.

Harris, Zellig S. 1952. Discourse analysis. *Language*, 28(1):1–30.

Harris, Zellig S. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Harris, Zellig S. 1957. Co-occurrence and transformation in linguistic structure. *Language*, 33(3):283–340.

Hill, Felix, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Hodgson, James M. 1991. Informational constraints on pre-lexical priming. *Language and Cognitive Processes*, 6(3):169–205.

Jarmasz, Mario and Stan Szpakowicz. 2004. Roget's thesaurus and semantic similarity. In Nicolas Nicolov, Kalina Bontcheva, Galia Angelova, and Ruslan Mitkov, editors, *Recent Advances in Natural Language Processing III: Selected Papers from RANLP 2003*. Benjamins, pages 111–120.

Jurgens, David and Keith Stevens. 2010. The S-Space package: An open source package for word space models. In *Proceedings of the ACL 2010 System Demonstrations*, pages 30–35, Uppsala, Sweden, July.

Kanerva, Pentii, Jan Kristoferson, and Anders Holst. 2000. Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*, volume 22, Philadelphia, United States, August.

Kiela, Douwe and Stephen Clark. 2014. A systematic study of semantic vector space model parameters. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, pages 21–30, Gothenburg, Sweden, April.

Lambiotte, Renaud, J-C Delvenne, and Mauricio Barahona. 2008. Laplacian dynamics and multiscale modular structure in networks. *arXiv preprint arXiv:0812.1770*.

Landauer, Thomas K. and Susan T. Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211.

Lapesa, Gabriella and Stefan Evert. 2014. A large scale evaluation of distributional semantic models: Parameters, interactions and model selection. *Transactions of the Association for Computational Linguistics*, 2:531–546.

Lapesa, Gabriella and Stefan Evert. 2017. Large-scale evaluation of dependency-based DSMs: Are they worth the effort? In *Proceedings of the 15th Conference of the European Chapter of the*

Association for Computational Linguistics: Volume 2, Short Papers, pages 394–400, Valencia, Spain, April.
Leech, Geoffrey Neil. 1992. 100 million words of English: the British National Corpus (BNC). *Language Research*, 28(1):1–13.
Lenci, Alessandro. 2008. Distributional semantics in linguistic and cognitive research. *Italian journal of linguistics*, 20(1):1–31.
Lenci, Alessandro. 2018. Distributional models of word meaning. *Annual review of Linguistics*, 4:151–171.
Lenci, Alessandro, Magnus Sahlgren, Patrick Jeuniaux, Amaru Cuba Gyllensten, and Martina Miliani. 2022. A comparative evaluation and analysis of three generations of distributional semantic models. *Language Resources and Evaluation*, 56:1269–1313.
Levy, Omer and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 302–308, Baltimore, United States, June.
Lewis, David D., Yiming Yang, Tony G. Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397.
Lin, Dekang. 1997. Using syntactic dependency as local context to resolve word sense ambiguity. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 64–71, Madrid, Spain, July.
Liu, Haitao, Chunshan Xu, and Junying Liang. 2017. Dependency distance: A new perspective on syntactic patterns in natural languages. *Physics of life reviews*, 21:171–193.
Lund, Kevin and Curt Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, & Computers*, 28(2):203–208.
Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, Baltimore, United States, June.
McDonald, Scott and Chris Brew. 2004. A distributional model of semantic context effects in lexical processing. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 17, Barcelona, Spain, July.
Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013*, Scottsdale, United States.
Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26:3111–3119.
Miller, George A. and Walter G. Charles. 1991. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.
Nelson, Douglas L., Cathy L. McEvoy, and Thomas A. Schreiber. 2004. The university of south florida free association, rhyme, and word fragment norms. *Behavior Research Methods, Instruments, & Computers*, 36(3):402–407.
Nivre, Joakim, Johan Hall, and Jens Nilsson. 2006. Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of the 5th International Conference on Language Resourcers and Evalutation (LREC 2006)*, volume 6, pages 2216–2219, Genova, Italy, May.
Padó, Sebastian and Mirella Lapata. 2007. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.
Peters, Matthew E, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, United States, June.
Rohde, Douglas, Laura Gonnerman, and David Plaut. 2006. An improved method for deriving word meaning from lexical co-occurrence. *Communication of the ACM*, 8(01).
Rohde, Douglas L.T. 2002. Methods for binary multidimensional scaling. *Neural Computation*, 14(5):1195–1232.
Rubenstein, Herbert and John B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Ruge, Gerda. 1992. Experiments on linguistically-based term associations. *Information Processing & Management*, 28(3):317–332.
Sahlgren, Magnus. 2005. An introduction to random indexing. In *Proceedings of Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering*, Copenhagen, Denmark, August.
Sahlgren, Magnus. 2008. The distributional hypothesis. *Italian Journal of Disability Studies*, 20:33–53.
Schober, Patrick, Christa Boer, and Lothar A. Schwarte. 2018. Correlation coefficients: Appropriate use and interpretation. *Anesthesia & Analgesia*, 126(5):1763–1768.
Schutze, Hinrich. 1992a. Dimensions of meaning. In *Supercomputing'92: Proceedings of the 1992 ACM/IEEE Conference on Supercomputing*, pages 787–796, Minneapolis, United States, November. IEEE.
Schütze, Hinrich. 1992b. Word space. In *Advances in Neural Information Processing Systems (NIPS Conference)*, volume 5, pages 895–902, Denver, United States. Morgan-Kaufmann.
Schutze, Hinrich and Jan O. Pedersen. 1995. Information retrieval based on word senses. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas, United States, April. Citeseer.
Strzalkowski, Tomek. 1994. Building a lexical domain map from text corpora. In *COLING 1994 Volume 1: The 15th International Conference on Computational Linguistics*, Kyoto, Japan, August.
Taieb, Mohamed Ali Hadj, Torsten Zesch, and Mohamed Ben Aouicha. 2020. A survey of semantic relatedness evaluation datasets and procedures. *Artificial Intelligence Review*, 53(6):4407–4448.
Turney, Peter D. 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of Machine Learning: ECML 2001*, pages 491–502, Freiburg, Germany, September. Springer.
Vietri, Simona. 2004. *Lessico-grammatica dell'italiano. Metodi, descrizioni e applicazioni*. Utet, Torino.
Wang, Yile, Leyang Cui, and Yue Zhang. 2021. Improving skip-gram embeddings using BERT. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1318–1328.
Yarowsky, David. 1992. Word-sense disambiguation using statistical models of roget's categories trained on large corpora. In *COLING 1992 Volume 2: The 15th International Conference on Computational Linguistics*, Nantes, France, August.