

Deriving Learning Strategies from Words Lists: Digital Dictionaries, Lexicons, Directed Graphs and the Symbol Grounding Problem

Jean-Marie Poulin and Alexandre Blondin Massé

Département d'informatique
Université du Québec à Montréal
Montréal, QC, Canada H3C 3P8

Email: {poulin.jean_marie, blondin_masse.alexandre}@uqam.ca

Abstract—We examine the structure of dictionaries, more specifically the interweaving of links that connect words through their definitions. With few exceptions, all the words used to construct dictionary definitions are defined somewhere else in the dictionary. All these references between words create a network of relations, thus making it possible to use graph theory for the study of dictionary structures. We propose using words learning as an investigative tool. For a given dictionary or lexicon, what would be the best strategy to learn all its headwords? To answer this question, we introduce a formal model and simple graph algorithms. We evaluate several different learning strategies by comparing their learning rate and their efficiency for 8 monolingual English-language dictionaries. It turns out that the most significant factor affecting the performance of learning strategies is their ability to break definitions circularity. In other words, the most effective learning strategies are the ones that break definition loops as quickly as possible. We show that a very simple algorithmic strategy, based solely on the vertices out-degree - the number of definitions in which lexemes participate - significantly improves the learning process when compared to psycholinguistic-based strategies. We also put forward that such an approach represents an efficient alternative for the construction of “word lists” used to teach foreign languages.

Keywords—Dictionaries; Lexicons; Learning Strategies; Word Lists; Graph theory.

I. INTRODUCTION

In this paper, which is an extended version of our earlier research presented at COGNITIVE 2018 [1], we examine the internal structure of dictionaries from a new and different perspective.

Whether in the form of clay tablets, papyrus, manuscripts, printed books, web pages or electronic tablets, dictionaries have existed for a very long time [2]. Since Antiquity, they have been commonly used as reference works in all areas of knowledge related to language. Even today, they are indispensable resources for reading, writing and translating texts, as well as for acquiring general knowledge.

With the advent of printing at the beginning of Renaissance, and even more so with the development of computers and the digital representation of knowledge in the 20th century, dictionaries underwent profound metamorphoses. In spite of this, dictionaries, lexicons and encyclopedias of all kinds still retain their relevance today. Open platforms, such as Wiktionary and Wikipedia, or Web versions of commercial dictionaries, such as Merriam-Webster [3] or Collins [4], are becoming ever more popular.

One of the key drivers of this success is undoubtedly the integration of hypertext and hyperlinks. These new technologies allow lexicographers to easily establish different types of links between words and concepts within a single publication, or even direct the user to external web resources. It then becomes possible to easily navigate from one word to another, without having to browse laboriously through the thousands of pages of a paper book. The way one uses dictionaries is thus profoundly modified. The relationships between the words become as important as the information about the words themselves.

Helped by developments in cognitive psychology and natural language processing, researchers have begun to question how these links between the words in dictionaries are organized. Are there invariants or schemes common to all dictionaries? This question was the main topic of several articles dealing with the structure of dictionaries.

In one of the first contributions on the subject, Clark [5] considered two special dictionaries: Longman’s Dictionary of Contemporary English (LDOCE) [6] and Cambridge’s International Dictionary of English (CIDE) [7]. LDOCE and CIDE have the particular feature of having a small *control vocabulary*, i.e., the number of words used in at least one definition is minimized. He showed that the words in the control vocabulary have distinctive properties: they are in general more abstract and their definition is longer and more complex than that of other words. Subsequently Steyvers and Tenenbaum [8] continued the analysis of the graphs associated with the WordNet Semantic Network [9] and Roget’s Thesaurus [10].

Pursuing the same topic, a group of researchers have recently published a series of articles exploring the internal structure of dictionaries [11]–[15]. These works all use a formal model for the lexicon based on graph theory. This approach makes it possible to apply conventional graph processing algorithms to dictionaries and derive a wealth of linguistically relevant information. Their analysis of several English-language digital dictionaries shows that all of them have a common structure and contain the same basic components [14], i.e.,

- A Kernel**, which is a subset of words in the dictionary that can be used to define all other words. The kernel can in turn be subdivided into a series of subcomponents of varying size, consisting of clusters of closely related words.
- A Core**, which is the subcomponent of the kernel with the largest number of words. In all the digital dictionaries

studied, the core is considerably larger than the other subcomponents of the kernel.

A Minimum Grounding Set (MinSet), which consists of a subset of words smaller than the core, obtained by judiciously combining elements of the core and other subcomponents of the kernel. This is the smallest group of words that can be used to define all the other words in the dictionary.

In addition, it turns out that the kernel has some distinct psycholinguistic characteristics [15]:

- The words in the kernel are learned earlier, are more concrete and are used more frequently than other words in the dictionary.
- There is a strong correlation between the evaluated psycholinguistic variables age of acquisition, degree of abstraction and words frequency.
- Within the kernel itself, there is a marked gradation of these same measures when evaluating words from the kernel, the core and the minimum grounding set.

Among these observations, the key point is most likely the minimum grounding set (Minset) question. In [15] the authors establish a direct link between the Minset and the "*Symbol Grounding Problem*", first described by Harnad in [16]. This problem can be briefly summarized as follows: When one looks for the definition of a word in a dictionary, he sees that this definition is built using other words. If these other words are not known, one can, of course, look for them in the dictionary. But, at the risk of getting caught in an endless loop, the meaning of some words must be known and rooted in the sensorimotor experience: "[...] it can not be dictionary lookups all the way down!" [15].

The symbol grounding problem is especially acute when learning a new language. In addition to getting familiar with grammar and syntax, one must acquire vocabulary and learn enough words to be able to understand and be understood. It must be possible to associate the external form of a written or spoken word with its meaning in a given context. According to Schmitt [17]: "[The] form-meaning link is the first and most essential lexical aspect which must be acquired".

What is the best way to learn these new words? Are there special learning methods or preferred strategies? In many research works, such as Prince [18], Schmitt [19], and Joyce [20], the authors compare traditional approaches used by instructors to teach second language learners. In the first method called "L1 translation", new English words are explained to the student in its mother tongue (the first language or L1). For example, if the student is Spanish-speaking, the teacher would give him an explanation of the English word *cat* in Spanish, i.e., *gato* or *felino*. With the second approach, named either "L2 context" or "L2 definition", the student must deduce by himself the meaning of a new word using the context in which it was seen or through some other explanation in English (the 2nd language or L2). One could for example explain to Jacques, a French-speaking student, the English word *own* with a definition such as "to have or hold as property". Joyce [20] compares these two methods. The "L1 translation" method is preferred for students with lower levels of proficiency: "[...] L1 translations for intentional vocabulary learning is seen to be most effective for students at lower proficiency levels". On the other hand, the "L2 definition"

approach is the most effective for vocabulary development: "for the purposes of general language development, learning through an L2 definition is favored".

A simple language dictionary can thus be a surprisingly effective way to understand and memorize new words. But for this to be successful, there is however an important prerequisite. The learner must first master a *basic subset* of the words in the new language. Only in this way will he be able to profitably use a dictionary.

Let us illustrate this point with the student Jacques in the previous example. Suppose Jacques sees in an English text the word *own*, which he does not know. He therefore consults the Merriam-Webster and finds the definition: "to have or hold as property" [21]. Assuming that he already knows the meaning of the words *to*, *have*, *or*, *hold* and *as*, but not that of the word *property*, he looks further in the dictionary and finds a definition for the word *property*: "something that is owned by a person". Although he is familiar with the words *something*, *that*, *is/be*, *by* and *person*, this definition is not useful for him. He faces what we call a definition loop: he needs to know the meaning of *owned/own* to understand the meaning of *property*, whereas at the beginning he was trying to understand this same word *own*. This is the difficulty we previously mentioned, the "*Symbol Grounding Problem*". Dictionary definitions are not enough by themselves to learn new words. In order to break out of the Kafkaesque situation created by the definition loop, one of these 2 words has to be learned some other way. In this case, Jacques could ask his teacher to explain him either the word *own* or the word *property*.

These same issues lie at the heart of our questioning. We aim to study the close relationship between the structure of monolingual dictionaries and the way in which the words of a language can be learned. In the references cited above (eg: [15]), the authors analyze the internal structure of dictionaries using groups of words with specific properties in terms of graph structure or psycholinguistic characteristics. They evaluate the definitional relations between the words to determine if it is possible to discover clusters of words having properties related to symbol grounding. Our approach is complementary. We first develop word lists, called *learning strategies*, based either on sequences of words coming from existing psycholinguistic norms, or built using graph theory algorithms. We then study the behavior of our *learning strategies* with respect to a reference task: "learning" all the words in a dictionary. We determine how effectively the strategies manage to break the definition loops in the dictionaries, thus avoiding the *symbol grounding problem*.

The rest of this document is organised as follows. In Section II, after having introduced some linguistic terminology and recalled basic notions of graph theory, we propose a convenient way to represent a lexicon as a directed graph. In Section III, we describe the notion of *learning strategy*. We first look in more detail at the problem of symbol grounding. Next, we discuss the question of word lists, these teaching tools frequently used by language instructors. Subsequently, we propose a formal learning model as well as related algorithms used to evaluate the strategies efficiency, regarding their ability to perform the task of "learning" all the words of a lexicon. We outline in Section IV our experimental environment and document the source of the digital dictionaries and psycholinguistic norms. Then we describe in detail the two types of

learning strategies developed:

- the algorithmic strategies, built using graph theory algorithms;
- the psycholinguistic strategies, based on psycholinguistic norms, i.e., lists of words ordered according to specific psycholinguistic properties.

Section V is devoted to the actual description of the experiments carried out. We present how we collected data and measured the performance of the strategies used to learn whole dictionaries. Then we outline the results obtained in the form of tables and graphs and offer a quick analysis of the most significant observations. Section VI completes our presentation by highlighting important findings and suggesting other avenues for future research.

It should also be noted that this article is a free French to English translation, with several modifications, of the first author's Master thesis [22].

II. DICTIONARIES, LEXICONS AND GRAPHS

In order to clearly position the subject of our study, let us look at some common definitions of the word "dictionary".

"Dictionary: a reference source in print or electronic form containing words usually alphabetically arranged along with information about their forms, pronunciations, functions, etymologies, meanings, and syntactic and idiomatic uses"

Merriam-Webster [3]

"Dictionary: a book that gives a list of words in alphabetical order and explains their meanings in the same language, or another language"

Longman [23]

These descriptions are consistent with the traditional view that most people hold. However, if we study them in more details, a central element of the definition stands out: the term *words*. In the following example, we look at 2 sentences where *word* is used with two different meanings.

Example 1.

- "Parce que" is a French word that translates to "because". In this sentence *word* refers to the whole "Parce que" group.
- "Parce que" is written in two words. In this case, *word* corresponds directly to the usual definition of a *word* as suggested by Jackson [24]: "a sequence of letters bounded by spaces".

Here is another example, showing another aspect of the ambivalence of *word*.

Example 2.

- We found a cat on the porch.
- There are many cats in the neighborhood.

Here the problem is a variant of the one in the previous example. Are *cat* and *cats* two different *words*? If we apply once again the definition from Jackson [24], we can infer that they are different *words*. But the fact is that in both cases we clearly refer to the same "small domestic animal known for

catching mice" [3]. In sentence 2 b), the plural form *cats* is used to show that we are talking about several animals.

This kind of ambiguity thus represents an important problem for our intended automated dictionary processing: the term *word* is not precise enough. We need to find a better way to distinguish its various uses. This is the reason why we first introduce a more precise linguistic terminology, allowing us to mitigate the imprecision of the vocabulary. We then put this terminology to work in order to propose a more formal definition of a lexicon. Thereafter, after having recalled some elementary notions of graph theory, we describe a way to represent a lexicon as a directed graph.

A. Terminology

There is no consensus amongst the different schools of linguistics as to which terminology is to be preferred. In this section we therefore propose, in order to simplify the understanding of our document, a list of the basic linguistic terms needed to describe our formal model.

Lexicon:

From a linguistic point of view, what is the difference between a lexicon and a dictionary? In English, the term lexicon is a common synonym for dictionary. According to the Merriam-Webster [25] or the Handbook of Linguistics [26], it is a book containing a list of words, accompanied by their definition, presented in alphabetical order. In our article, we use the term lexicon in its strict linguistic sense, namely: "the theoretical entity that corresponds to all the lexical items of a language or of an individual, i.e., the mental lexicon" [27, p. 109]. Note that this definition refers to a "set of lexical items", and not to a "set of words". To further highlight the difference between a dictionary and a lexicon, let us add a few precisions:

- 1) A dictionary is a model, a particular representation of a language's lexicon. It emphasizes the descriptive aspect, the definition of the words.
- 2) A dictionary is usually presented in alphabetical order, while there is no such imperative for a lexicon.
- 3) In a lexicon, the relationships between words are as important as the words themselves: it is not just a sequential list of words. One can also see a lexicon as a web of words linked together by a complex network of various relationships.

Amongst the many different relationships that words can have between them, let us look at a few examples:

Example 3.

- In the sentence: "The cat is a domestic animal", CAT and ANIMAL are connected to each other by relations of hyponymy and hyperonymy. CAT is a hyponym of ANIMAL, while in the opposite direction, ANIMAL is a hyperonym of CAT.
- In the sentence "I saw a stray cat", the words CAT and STRAY are connected by another type of relationship. STRAY is a quality that is commonly applied to a CAT. However, the qualifier PURPLE, as in "I saw a purple cat", is mostly inappropriate for a cat, unless used in a very specific context, like in a comic book.
- If we define a CAT as a "small domestic animal known for catching mice" [3], the *words* SMALL, DOMESTIC,

ANIMAL, etc., have here a different relationship with CAT. They help to describe, to define what a CAT is.

Later on, we use this last type of relationship, termed a “definitional relation”, to explore the structure of lexicons.

Words, lexemes and others:

Let us look now at the different elements that make up our terminology. Figure 1 illustrates, in the form of an entity-relationship diagram, the reciprocal links that unite the linguistic terms required for our analysis. These terms, as well as the associated writing conventions, are strongly inspired by Polguère [27], [28].

word form: The Oxford Dictionary defines a **word form** as: “a (particular) form of a word; especially each of the possible forms taken by a given lexeme, distinguished by their grammatical inflections” [29].

Without going further into linguistic theory, we simply say that *cat* and *cats* are two different word forms of the lexeme CAT, both of which refer to the same lexical meaning <cat>. The terms “lexeme” and “lexical meaning” are defined later on.

Writing convention:

A **word form** is noted in italics, for example *cats*.

lexical item: A **lexical item** - or headword - is the basic unit of a lexicon, equivalent to an entry in a dictionary. “A lexical item, also called a lexical unit, is either a lexeme or a phrase. Each lexicon (lexeme or phrase) is associated with a given meaning [...]” [27, p. 69]

For example, “seat belt” and “cat” are both lexical items. “cat” is a simple lexical item consisting of a single word-form, equivalent to the lexeme CAT. On the other hand, “seat belt” is a compound lexical item comprising 2 associated word forms.

In our analysis however, we do not tackle the task of deciding whether a group of words corresponds to a compound lexical item or not. We believe that is a different, quite difficult problem, worthy of consideration on its own. We thus consider further on all word-forms as candidate lexemes.

lexeme: Let us examine again the two sentences in example 2. We understand that the two word-forms *cat* and *cats* both make reference to the same concept or idea: the lexical meaning <cat>. These word-forms are simply “inflected forms” of the same **lexeme** CAT. Here, “inflected form” refers to a morphological change, the addition of an affix or special ending to the final of a word (noun, pronoun, participle, adjective) according to its function in the sentence or proposition [30].

Polguère [27] defines a lexeme as a generalization of word-form linguistic signs: each lexeme of the language is structured around a meaning that can be expressed by a set of distinct word-forms. In other words, we can think of a lexeme as a way of identifying a precise lexical meaning, to which a series of grammatical variations represented by the different word-forms are associated. In the same manner, the word-forms *write*, *writes*, *written*, ... are different grammatical forms of the same lexeme WRITE (Spencer [31]).

Writing convention:

Lexemes are written in small capital letters, as in CAT. They can also be tagged, as in $CHAT_N^1$, where the exponent “1” indicates the result of disambiguation and the index “N” represents the part of speech.

lexical meaning: In this paper, we use the term **lexical meaning** to refer to the idea, the mental representation, to which a lexeme refers. “The lexical meaning refers to a mental concept that is associated with a lexical unit to express an idea” [32] The term lexical meaning can, according to the disciplines and the authors, be put in parallel with the related notions of concept: “Between all the individuals thus connected by the language, it will establish a kind of average: all will reproduce [...] the same signs united to the same concepts” [33], as well as category in philosophy and cognitive psychology, and signified in semiotics.

Writing convention:

The lexical meaning of a lexeme is noted with chevrons. For example, <cat> is the lexical meaning associated with the lexeme CAT.

lemma: According to Polguère, a **lemma** is the canonical word form used to designate a term [27, p. 135]. In French for example, we use the infinitive present to represent a verb, the masculine singular to represent a noun, etc. According to our nomenclature, we say that it is the word-form that has been chosen to identify one or more lexemes.

Writing convention:

1. A lemma is written in non-proportional font, for example CAT.
2. To distinguish the lexemes associated with the same lemma, we use an exponent between 1 and n , for example $CHAT^1$, $CHAT^2$, ..., $CHAT^n$.

In automatic language processing, lemmatization is the operation consisting of identifying the lemma that corresponds to the different word-forms of a lexeme. For example: the lemma GO is the result of the lemmatization of the word-forms *goes*, *went*, ... The distinction between the terms “lexeme” and “lemma” can sometimes seem difficult to establish. To help make them stand out, one only needs to remember that lexeme is rather related to meaning, to semantics, whereas lemma is related to form, to morphology.

part of speech: The parts of speech (POS) are classes that group together lexical items according to their grammatical properties [27]. For the purposes of our presentation, we consider that all lexemes are part of exactly one of the following 5 parts of speech: *noun*, *verb*, *adjective*, *adverb* and *stop word*. The first four classes - *noun*, *verb*, *adjective* and *adverb* - group the vast majority of lexemes. The fifth class, *stop word*, groups all the other lexemes whose semantic value is poorer.

Writing convention:

The part of speech of a lexeme or a lemma is represented by a coding label: “N” for name, “V” for verb, “A” for adjective, “R” for adverb and “S” for stop word. For example, CAT_N indicates that the lexeme CAT is a noun.

In natural language processing (NLP), the term “lemma-

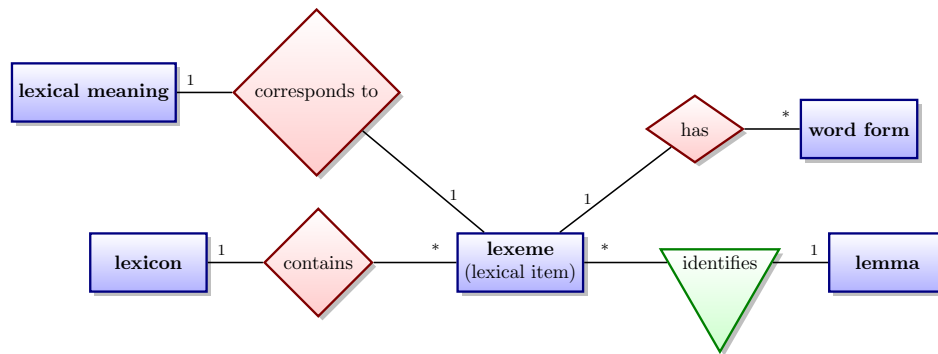


Figure 1. Entity-relationship diagram of linguistic terms (simplified)

tization” is used to refer to this process of identifying the part of speech to which the lexemes of a sentence or text belong.

Polysemy and Disambiguation:

We previously mentioned that a given lemma can correspond to more than one lexeme. In this regard, linguistic researchers usually make the distinction between two different situations [34] :

- *homonymy*, when the lexemes are of different etymological origin.
- *polysemy*, when the lexemes refer to different meanings of the same word

The next example, extracted from Jurafsky and Martin’s work [35], illustrates in a classical way how a “word” can have many different meanings:

Example 4.

- “Instead, a *bank* can hold the investments in a custodial account in the client’s name.”
- “But as agriculture burgeons on the east *bank*, the river will shrink even more.”
- “The *bank* is on the corner of Nassau and Witherspoon.”

To understand these sentences, one has to choose amongst some possible meanings for the lexeme BANK which one is the most appropriate, for example:

- $BANK_N^1$: “financial institution”,
 $BANK_N^2$: “building belonging to a financial institution”,
 $BANK_N^3$: “sloping mound”,

In sentences (a) and (c), the context is quite different. It is relatively easy to disambiguate BANK. Sentence (a) is about investment, account and client, so $BANK_N^1$ is the most appropriate. In (b) however, $BANK_N^3$ is the most relevant since we are in the context of agriculture, river, etc. Given that the semantic domain is downright different, we can easily identify them as homonyms. On the other hand, sentence (c) is more difficult to analyze. We do not have many clues from the context to guide our choice. One must know or figure out that Nassau and Witherspoon are street names and then infer

that we are talking about a building, therefore the branch of a bank. $BANK_N^2$ and $BANK_N^1$ are thus polysemous.

This complex process of discriminating the meaning of words is called “Word Sense Disambiguation” (WSD) or simply disambiguation. For a human, the distinction is made naturally, without apparent effort. It is however much more difficult for an algorithm or computer program: “The reason that lexical polysemy causes so little actual ambiguity is that, in actual use, context provides information that can be used to select the intended sense. Although contextual disambiguation is simple enough when people do it, it is not easy for a computer to do” [36] According to Corrêa, Lopes and Amancio, the question of lexical disambiguation in artificial intelligence even remains an unresolved problem in 2018 [37]. For several authors, it is considered an AI-complete problem. In other words, by analogy with NP-complete problems in complexity theory, it is a problem as difficult as the creation of a real artificial intelligence [38], [39].

There is thus no cost-efficient and reliable way to disambiguate the meaning of words in a sentence. However, when they build or revise dictionaries, lexicologists usually order word senses according to their usage frequency, starting from the most frequently used : CIDE: [7, p. ix], LDOCE: [40], WORDSMYTH: [41]. Thus, by simply using the definitions order, “the heuristic of the first sense” generally gives satisfactory results. This method is still a baseline difficult to surpass: “The first sense heuristic [...] outperforms many of these systems which take surrounding context into account” [42]. For these reasons, as well as for the sake of simplicity, we use the first sense heuristic as a disambiguation method in this work.

B. Formal definition of a lexicon

As we have seen above, a lexicon can be described from a linguistic point of view as a set of lexemes accompanied by their definitions and any other information necessary for their use [26].

However, for our analysis, we need to go further in terms of mathematical formalism. Proceeding by successive refinements, we propose in this section the formal definition of a *complete lexicon*.

Definition 1 (Lexicon). A *lexicon* is a quadruple $X = (A, \mathcal{P}, \mathcal{L}, \mathcal{D})$, where:

- (i) \mathcal{A} is an alphabet, whose elements are called letters.
- (ii) $\mathcal{P} = \{N, V, A, R, S\}$ is a non-empty set of elements called *part of speech* (POS). The elements correspond to the 5 parts of the speech described earlier.
- (iii) \mathcal{L} is a finite set of triplets $\ell = (w, i, p)$, called *lexemes* and denoted $\ell = w_p^i$, where $w \in A^*$ is a word form, $i \geq 1$ is an integer, and $p \in \mathcal{P}$. We then say that (w, i, p) is the i -th sense of the tagged word form (w, p) :
- If there is no $(w, i, p) \in \mathcal{L}$ with $i > 1$, then $w_p^1 \equiv w_p$ and (w, p) is *monosemic*. Moreover, if all $(w, i, p) \in \mathcal{L}$ are monosemic, then we say that X is monosemic.
 - If there exists a $(w, i, p) \in \mathcal{L}$ with $i > 1$, we say that (w, p) and X are *polysemic*.
 - To make the numbering consistent, we assume that if $(w, i, p) \in \mathcal{L}$ and $i > 1$, then $(w, i-1, p) \in \mathcal{L}$ is also true.
 - If $p = S$, then $\ell = w_s^i$ is called a *stop lexeme*.
- (iv) \mathcal{D} is a function that associates with each lexeme $\ell \in \mathcal{L}$ a finite sequence $D(\ell) = (d_1, d_2, \dots, d_k)$, where $d_i \in A^*$ for $i = 1, 2, \dots, k$. It is called the definition of ℓ .

We can see in Example 5 a polysemic lexicon.

Example 5. Let $X = (\mathcal{A}, \mathcal{P}, \mathcal{L}, \mathcal{D})$ be a lexicon such that:

- $\mathcal{A} = \{a, b, \dots, z\}$
- $\mathcal{P} = \{N\}$, where N shows that the part of the speech is a NOUN,
- \mathcal{L} and \mathcal{D} are as defined in Table I.

TABLE I. Lexemes and definitions of a polysemic lexicon

ℓ	$D(\ell)$
FRUIT _N ¹	(plant, part, that, has, seed, and, edible, flesh)
FRUIT _N ²	(the, result, of, work, or, action)
FLESH _N ¹	(the, edible, part, of, a, fruit, or, vegetable)
FLESH _N ²	(the, part, of, an, animal, used, as, food)
SEED _N ¹	(the, small, part, of, a, plant, from, which, a, new, plant, can, develop)

Definition 2 (Lemmatized lexicon). Let $lemma(w)$ be a function that associates to a word-form $w \in A^*$ its lemma. If we replace in Definition 1 (iv) $D(\ell) = (d_1, d_2, \dots, d_k)$ by $D(\ell) = (lemma(d_1), lemma(d_2), \dots, lemma(d_k))$, then $D(\ell)$ is called a *lemmatized definition* of ℓ .

We then say that X is a *lemmatized lexicon*.

Definition 3 (Tagged Lexicon). If we replace in Definition 2 (iv) the condition $d_i \in A^*$ with $d_i \in \mathcal{A}^* \times \mathcal{P}$, then $D(\ell)$ is called a *tagged definition* of ℓ .

We then say that X is a *tagged lexicon*. Example 6 shows such a lexicon.

Example 6. Let $X = (\mathcal{A}, \mathcal{P}, \mathcal{L}, \mathcal{D})$ be a lexicon such that:

- $\mathcal{A} = \{a, b, \dots, z\}$
- $\mathcal{P} = \{N, V, S\}$, where $N \rightarrow$ NOUN, $V \rightarrow$ VERB, $S \rightarrow$ STOP
- \mathcal{L} and \mathcal{D} are as defined in Table II.

TABLE II. Lexemes and definitions for a tagged lexicon

ℓ	$D(\ell)$
HAVE _V	(to _S , own _V , or _S , possess _V)
OWN _V	(to _S , have _V , in _S , your _S , possession _N)
POSSESS _V	(to _S , have _V , in _S , its _S , possession _N , to _S , own _V)
POSSESSION _N	(having/have _V , or _S , owning/own _V , something _S)

Definition 4 (Disambiguated lexicon). If we replace in Definition 3 (iv) the condition by $d_i \in \mathcal{L}$, then $D(\ell)$ is called a *disambiguated definition* of ℓ .

We then say that X is a *disambiguated lexicon*.

Definition 5 (complete lexicon). Finally, if X is a *disambiguated lexicon* such that for every ℓ that is not a stop lexeme there is a $D(\ell)$, we then say that X is a *complete lexicon*.

Example 7. Let $X = (\mathcal{A}, \mathcal{P}, \mathcal{L}, \mathcal{D})$ be a lexicon such as:

- $\mathcal{A} = \{a, b, \dots, z\}$
- $\mathcal{P} = \{N, V, S\}$, where $N \rightarrow$ NOUN, $V \rightarrow$ VERB, $S \rightarrow$ STOP
- \mathcal{L} and \mathcal{D} are defined in Table III.

TABLE III. Lexemes and definitions for a complete lexicon

ℓ	$D(\ell)$
HAVE _V	(TO _S , OWN _V , OR _S , POSSESS _V)
OWN _V	(TO _S , HAVE _V , IN _S , YOUR _S , POSSESSION _N)
POSSESS _V	(TO _S , HAVE _V , IN _S , ITS _S , POSSESSION _N , TO _S , OWN _V)
POSSESSION _N	(having/HAVE _V , OR _S , owning/OWN _V , SOMETHING _S)

C. Graphs

In this section, we give a brief overview of the mathematical model used for our analysis of the structure of lexicons: the graph theory. But first, let us introduce the notion of semantic network.

For many authors specializing in artificial intelligence, a semantic network is an especially useful form of knowledge representation [43]–[45]. Lehmann gives a very concise definition: “A semantic network is a graph of the structure of meaning” [46]. In its traditional form, a semantic network represents objects in the form of nodes, connected to each other by links, which are optionally labeled. Figure 2 provides an example of a simple semantic network. Nodes and arrows represent a subset of a database of free associations [47]. In this study, the authors asked participants, after showing them a word, to name the first word that spontaneously came to their mind. For example, in the diagram in Figure 2, “volcano” is connected to “explode” by an arrow. This means that several participants spontaneously associated the word “explode” with the word “volcano” when the latter was used as a primer.

Using the same type of representation, one can easily imagine representing a lexicon as a graph where the lexemes are displayed as nodes and the relations between the lexemes are indicated by links between the nodes. As an example, let us go back to the definition of the lexeme HAVE_V in example 7:

$$D(\text{HAVE}_V) = (\text{TO}_S, \text{OWN}_V, \text{OR}_S, \text{POSSESS}_V)$$

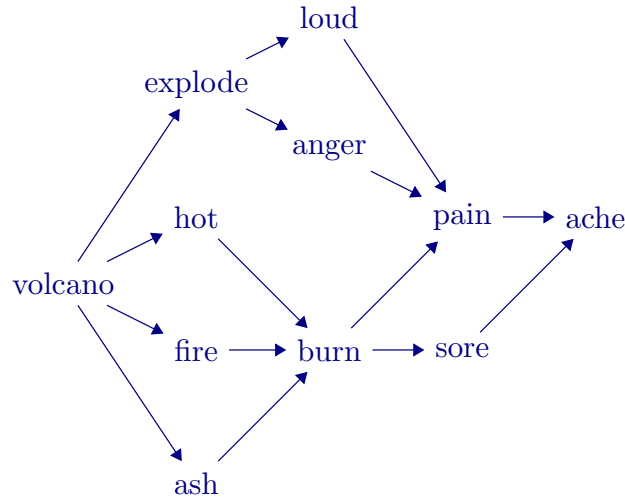


Figure 2. Association network [8].

Figure 3 represents this same definition of HAVE_v as a semantic network.

This form of representation is quite similar to the way Bondy and Murphy introduce the notion of graph [48, p. 1], i.e., “[...] a diagram consisting of a set of points together with lines joining certain pairs of these points”. But to properly represent definitional relations between lexemes, we must be more specific and introduce the notion of directed graph.

Definition 6 (Directed graph). A directed graph D – digraph – is an ordered pair (V, A) where:

- (i) V is a finite set of vertices,
- (ii) $A \subseteq V \times V$ is a finite set of elements called arcs,

Note: If $v_1, v_2 \in V$, then $(v_1, v_2) \in A$ does not imply that $(v_2, v_1) \in A$.

Example 8. Let $D = (V, A)$ be a directed graph with

$$V = \{v_1, v_2, v_3, v_4\},$$

$$A = \{(v_2, v_1), (v_3, v_1), (v_1, v_2), (v_3, v_2), (v_4, v_3), (v_2, v_3), (v_2, v_4), (v_1, v_4), (v_3, v_4)\}$$

Figure 4 is a visual representation of the digraph D .

From this definition of a directed graph, we derive the following related notions:

degree

Let $D = (V, A)$ be a directed graph. For $u, v \in V$, u is a predecessor of v if $(u, v) \in A$. The set of predecessors of v is written $N^-(v)$. The number of predecessors of v is called the *in-degree* of v , represented by $\text{deg}^-(v)$. In the same manner, we say that v is a successor of u if $(u, v) \in A$ and that the set of successors of u is denoted $N^+(u)$. In this case $\text{deg}^+(u) = |N^+(u)|$ is called the outer degree of u .

circuit

A finite sequence $p = (v_1, v_2, \dots, v_k) \in V^k$ is called a

path of D if $(v_i, v_{i+1}) \in A$ for $i = 1, 2, \dots, k - 1$. If in addition $v_1 = v_k$, then p is called a circuit.

feedback vertex set

A feedback vertex of D is a subset $U \subseteq V$ of vertices such that, for any set of vertices c forming a circuit in D , the set $U \cap c$ is non-empty [49]. That is, U covers all circuits of D . The *minimum feedback vertex set (MFVS) problem* consists in finding in a graph a feedback vertex set of size as small as possible. For a general graph, it is an NP-hard problem, namely that there is no algorithm to solve this problem in polynomial time unless $P = NP$ [50]. However, by using combinatorial operators and linear programming techniques [51], [52], Vincent-Lamarre et al. [15] have succeeded in solving the problem for the smallest lexicons they considered and in finding a good approximation for the other ones.

strongly connected component

For $u, v \in V$, we write $u \rightarrow v$ if there exists a path from u to v and we write $u \leftrightarrow v$ if both $u \rightarrow v$ and $v \rightarrow u$ hold. A strongly connected component (SCC) is a subgraph of D induced by an equivalence class of the relation \leftrightarrow over V . In other words, when it is possible to move from a vertex u to a vertex v in a strongly connected component, it is also possible to go in the opposite direction from the vertex v to the vertex u . Moreover, since \leftrightarrow is an equivalence relation and in particular, transitive, the induced subgraph will be of maximal size [11].

D. Lexicons and Associated Graphs

Directed graphs are especially suitable for representing the relations between the lexemes of a lexicon. For our analysis of the structure of lexicons, we consider only the definitional relations of the type: lexeme l “is part of the definition” of lexeme l' .

We represent a lexicon using the following conventions:

- The vertices of the graph correspond to the lexemes.

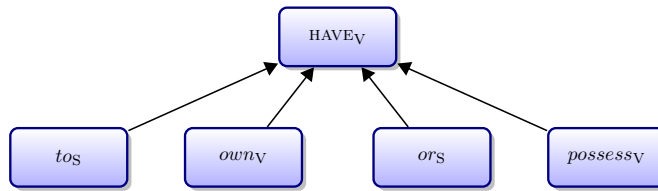


Figure 3. Semantic Network Representing Definitional Relationships

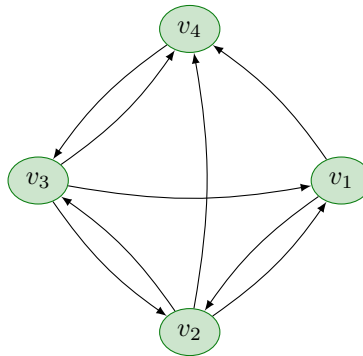


Figure 4. Digraph D

- The arcs between the vertices correspond to the relations between the lexemes. For example, if an arc goes from vertex l to vertex l' , it means that lexeme l is part of the definition of l' .
- With regard to stop lexemes, we consider that their lexical value is very low compared to lexemes of other parts of speech (noun, verb, adjective and adverb). We do not represent them in the associated graphs and we do not take them into account in our analysis. This way of doing things is used very often in NLP [35], in information research (RI) [53], and in data mining [54].

More formally, we define an “associated graph”, as follows.

Definition 7 (Associated graph).

Let $X = (\mathcal{A}, \mathcal{P}, \mathcal{L}, \mathcal{D})$ be a complete lexicon. Then $G(X)$ is X 's associated graph if:

- (i) $G(X) = (V, A)$ is a directed graph
- (ii) $V = \mathcal{L}$
- (iii) If $l \in D(l')$ and l is not a stop lexeme, then $(l, l') \in A$

The following example 9 shows the graph associated with the small lexicon X_{small} , containing 4 vertices – 4 lexemes – and 9 arcs – 9 definitional relations –.

Example 9.

Let $X_{small} = (\mathcal{A}, \mathcal{P}, \mathcal{L}, \mathcal{D})$ be a complete lexicon where \mathcal{L} and \mathcal{D} are shown in Table IV.

Figure 5 illustrates the graph corresponding to the lexicon X_{small} .

Example 10.

Figure 6 shows the graph associated with the larger $X_{large} = (\mathcal{A}, \mathcal{P}, \mathcal{L}, \mathcal{D})$ lexicon, comprising 40 vertices and

TABLE IV. Complete lexicon

ℓ	$D(\ell)$
HAVE _V	(TO _S , OWN _V , OR _S , POSSESS _V)
OWN _V	(TO _S , HAVE _V , IN _S , YOUR _S , POSSESSION _N)
POSSESS _V	(TO _S , HAVE _V , IN _S , ITS _S , POSSESSION _N , TO _S , OWN _V)
POSSESSION _N	(<i>having</i> /HAVE _V , OR _S , <i>owning</i> /OWN _V , SOMETHING _S)

123 arcs.

III. LEARNING STRATEGIES

In this section, we examine the relationship between the learning of new words and the structure of dictionaries.

First, we look into what is implied by the phrase “learning new words”. We seek to understand how we learn to associate a linguistic token, whether read or heard, with a meaning. For this purpose, we reexamine the symbol grounding problem [16] and the pedagogical approach traditionally used to mitigate this difficulty: the construction of word lists.

In a second step, we propose a high-level model to represent the process of learning new words. After having formally defined what “learning a new word” means in our context, we propose different algorithms to simulate this behaviour.

A. Learning new words

In this section, we address the issue of vocabulary acquisition, particularly in the context of second-language learning. First, we seek to identify the main difficulties encountered when using a monolingual dictionary to learn the meaning of the new words encountered.

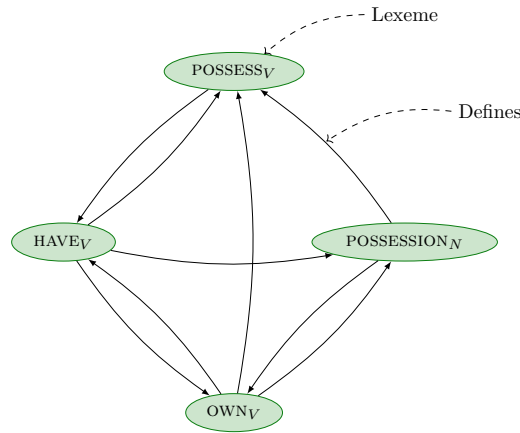


Figure 5. Graph associated with the lexicon X_{small}

Symbol Grounding Problem:

In several articles dealing with this matter, Harnad analyzes the problem of grounding symbols, the famous Symbol Grounding Problem [16], [55], [56]. Without going into the details of linguistics and cognitive science, this question can be summarized as follows: Where does the meaning of words come from? How is it that a word we know usually conjures up something specific? According to Harnad, this is because the words are grounded in a sensorimotor way:

“How are word meanings grounded? Almost certainly in the sensorimotor capacity to pick out their referents.” [56]

However, it is clear that the learning of new words does not occur in the same way for a young child assimilating the first basics of his mother tongue as for an adult studying a new language.

When a second-language learner encounters a word he does not know, one way around this difficulty may be to consult a dictionary to find the definition of the unknown word. If everything goes well, the definition allows him to “learn” the new word and memorize it. Let us illustrate this situation with an example from [16]:

- (1) Suppose a learner already knows the word *horse*, which is well grounded in his sensorimotor experience. He can easily recognize a horse if he sees one.
- (2) Let’s also suppose that *striped* is known in the same manner
- (3) He would then presumably be able to identify a zebra if he sees one, using only a simple definition such as “*striped horse*”. He could associate the symbol – the word *zebra* – with the animal that looks like a horse and that is striped.

But things get more complicated if there are too much words in the definition that he does not know. In the article of Blondin Massé *et al*, the authors describe the uncomfortable situation where one would endlessly run through the dictionary, going from unknown words to other unknown words, without hope of arriving at understanding of the words and of their definitions [11]. Therefore, for the definition of a word in a dictionary to be understandable and useful, a sufficient number

of words must already be “grounded”, that is, they must mean something more than abstract forms on paper or on a screen. We do not study further how words are actually grounded in sensorimotor experience. We keep in mind that if enough words in the definition are known and well grounded, one can learn a new word and ground it in turn.

Minimum Grounding Set:

We now examine how our formal model for lexicons and associated graphs reflects the symbol grounding issue.

First, assume that we can learn a new lexeme only if we already know all the lexemes that appear in its definition. We can then define a grounding set as a subset of lexemes allowing us to learn all the other lexemes in this lexicon.

Definition 8 (Grounding set). Given

- (a) $X = (\mathcal{A}, \mathcal{P}, \mathcal{L}, \mathcal{D})$ a complete lexicon,
- (b) $G(X) = (V, A)$ its associated graph,
- (c) $U \subseteq V$ a subset of V ,
- (d) L a function defined by $L(U) = U \cup \{v \in V \mid N^-(v) \subseteq U\}$,

if there is a $k \in \mathbb{Z}^+$ such that $L^k(U) = V$, we then say that U is a *grounding set* of X and that \mathcal{L} is *k-reachable*.

Looking again at lexicon X_{large} in Figure 6, we can use this definition to validate if a subset of the vertices of X_{large} is a grounding set.

Example 11.

Let us use a starting subset $U = \{ HAVE_V^1, PLACE_N^1, POSSESSION_N^1, QUALIFY_V^1, REFER_V^1, STATE_N^1, THING_N^1 \}$.

If we recursively apply the previously defined function L , we get:

$$\begin{aligned}
 L^0(U) &= U \\
 L^1(U) &= L^0(U) \cup \{ PARTICULAR_A^1, POSITION_N^1, OWN_V^1 \} \\
 L^2(U) &= L^1(U) \cup \{ POSSESS_V^1, SOMETHING_N^1 \} \\
 L^3(U) &= L^2(U) \cup \{ CONDITION_N^1 \} \\
 L^4(U) &= L^3(U)
 \end{aligned}$$

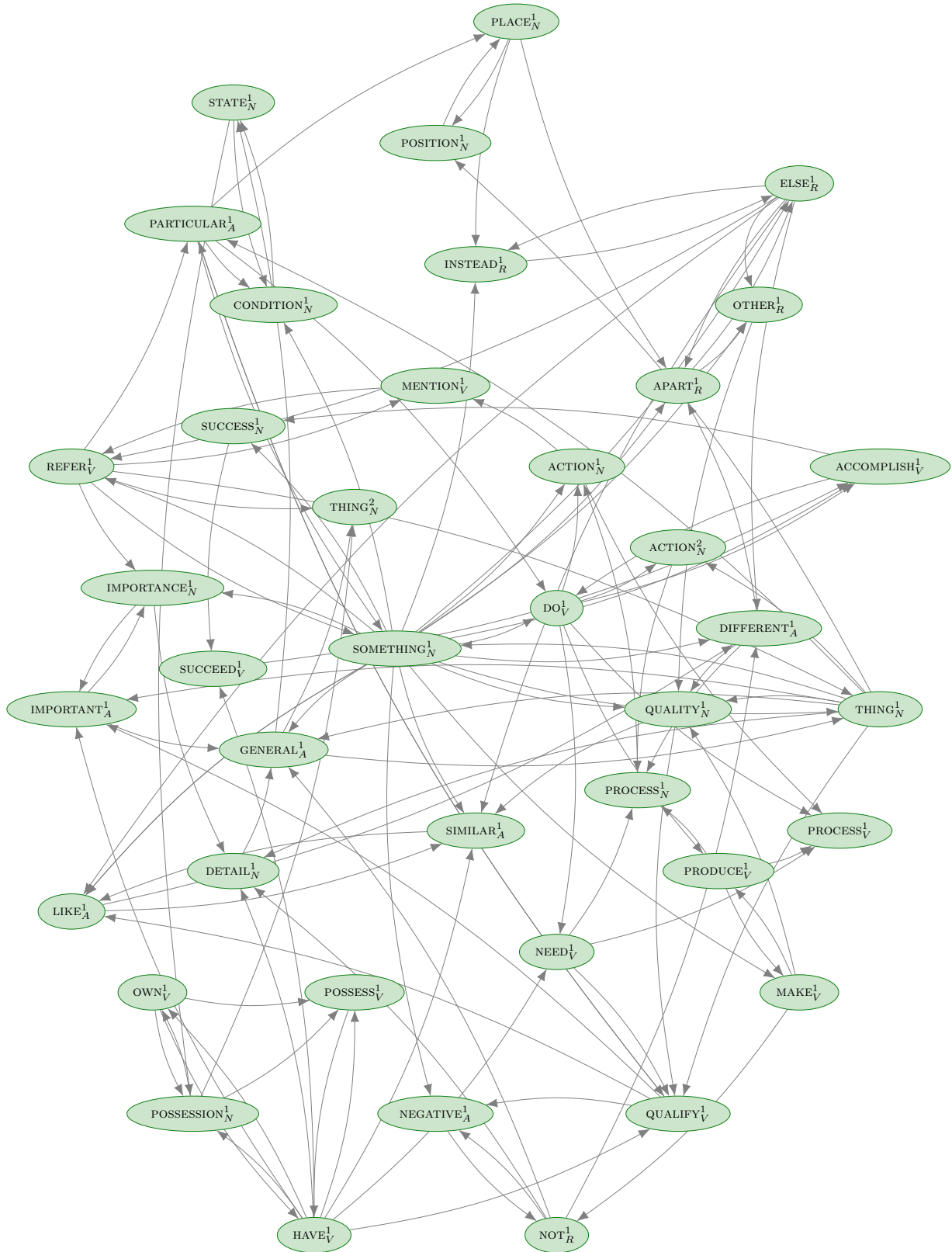






Figure 6. Graph associated with lexicon X_{large}

In Figure 7, the elements of the sets $L^0(U)$, $L^1(U)$, $L^2(U)$ and $L^3(U)$ are respectively marked with the symbols , ,  and .

For example, we can see that the lexeme OWN_V^1 is 1-reachable since it can be learned from the lexemes POSSESSION_N^1 and HAVE_V^1 . Similarly, the lexeme POSSESS_V^1 is 2-reachable since it can be learned from the lexemes POSSESSION_N^1 and HAVE_V^1 , and OWN_V^1 . Moreover, since we have $L^4(U) = L^3(U)$, there is no way we can learn additional lexemes and U is not a grounding set of the X_{large} lexicon.

Blondin Massé *et al.* have shown that there is also an exact correspondence between the grounding sets of a lexicon X and the feedback vertex sets of the associated graph $G(X)$ [11]. So, if U is a grounding set of $G(X)$, it means that we can learn by definition all the other lexemes of X , that is $V \setminus U$. As explained earlier in Section II-C, the calculation of a minimum feedback vertex set is, in general, an NP-hard problem. However, as the article by Vincent-Lamarre *et al.* [15] demonstrates, it is possible to use algorithms and linear programming techniques to calculate an exact solution or at worst to find a close-enough approximation. To illustrate the calculation of minimal grounding sets, let us look again at examples of complete lexicons presented in Section II-D.

Example 12.

For the trivial lexicon X_{small} from Figure 8, one can easily find by trial and error a minimum feedback vertex set, for instance: $\{\text{HAVE}_V^1, \text{POSSESSION}_N^1\}$.

Example 13.

On the other hand, for the X_{large} lexicon, which is still diminutive compared to a real-life dictionary, we find that the “manual” method is not adequate for finding a minimum grounding set. Figure 9 illustrates a minimum grounding set obtained using the method described in Vincent-Lamarre *et al.* [15].

$$\{\text{ACCOMPLISH}_V^1, \text{HAVE}_V^1, \text{IMPORTANT}_A^1, \text{LIKE}_A^1, \\ \text{MAKE}_V^1, \text{PLACE}_N^1, \text{POSSESSION}_N^1, \text{QUALIFY}_V^1, \\ \text{REFER}_V^1, \text{STATE}_N^1, \text{THING}_N^1, \text{NOT}_R^1, \text{ELSE}_R^1\}$$

B. Word lists

Let us look at the connection that can be established between the notions of symbol grounding and minimal grounding set, and the techniques used for teaching languages.

The importance given to vocabulary teaching in second language classes has varied over the years, following the evolution of theories and approaches in language didactics [57]. But the fact remains that for students, the acquisition of a large vocabulary is essential for attaining proficiency in a language. Teachers and researchers in applied linguistics have thus long sought ways to facilitate the learning of new words for their students. In this context, one can understand their interest in word lists.

Word lists are of word groupings representative of a specialized field or a language that students must master as early as possible to become autonomous in their study. They then have a base of known words allowing them to independently use dictionaries or other tools to help learning. According to Nation, “Word lists lie at the heart of good vocabulary course design” [58].

In the 1930s, Charles Ogden first introduced his “Basic English”, a version of English with simplified grammar and vocabulary [59]. Basic English was to become, according to Ogden, a universal language, somewhat like Esperanto. To facilitate the learning of this basic English, several lists of words - between 850 and 2000 words - were subsequently built [60].

In the 1950s, West proposed its General Service List (GSL), containing about 2000 words frequently used in English [61]. The GSL has since become a key reference: “There has been no comparable replacement for the GSL up to now” [62].

More recently, Brezina and Gablasova [63], as well as Browne [64] both proposed an improved version of the GSL, named in both cases the New General Service List (NGSL). Browne also suggests 3 additional lists to complement the NGSL [65]:

- The “New Academic Word List” (NAWL);
- The “TOEIC Service List” (TSL);
- The “Business Service Lists” (BSL).

But how are these lists constructed? The most commonly used method is to count the relative frequency of words in a collection of relevant documents and then classify those words

in a list according to their frequency and their importance for the author. In a recent publication, Nation presents a detailed description of list construction techniques using corpora [58].

In this article we propose a different approach, never used before as far as we know. With this new method, we use a lexicon and simple graph theory algorithms to efficiently build word lists. To accomplish this, we first represent the lexicon as a directed graph and then use graph algorithms to identify a list of words allowing us to effectively “learn” all the other words of the lexicon.

C. Learning model

In an article by Picard *et al.* [13], the authors put forward the hypothesis that there are two ways to learn new words or new lexical meanings: verbal instruction and direct sensorimotor induction.

We rely on this premise to build our formal model of learning. We say that a new lexeme can be learned in two different ways:

Direct learning: With this approach, lexeme and lexical meaning are directly connected through sensorimotor experience. For example, during a visit to a farm, someone could explain to a child that the animal in front of him is called a “horse”.

To keep our model simple, we do not concern ourselves about the way this link is established or what is going on at the mental and sensorimotor levels. We stick to the fact that it is a complex operation, which often requires the intervention of a person or some other entity to clarify the matter. We have to get out of the pure “world of words”, so we consider it to be a relatively costly process.

Definition learning: In this case, some lexical information is used to establish the link between the meaning and the lexeme; for example, a student searches a dictionary to find the definition of a zebra.

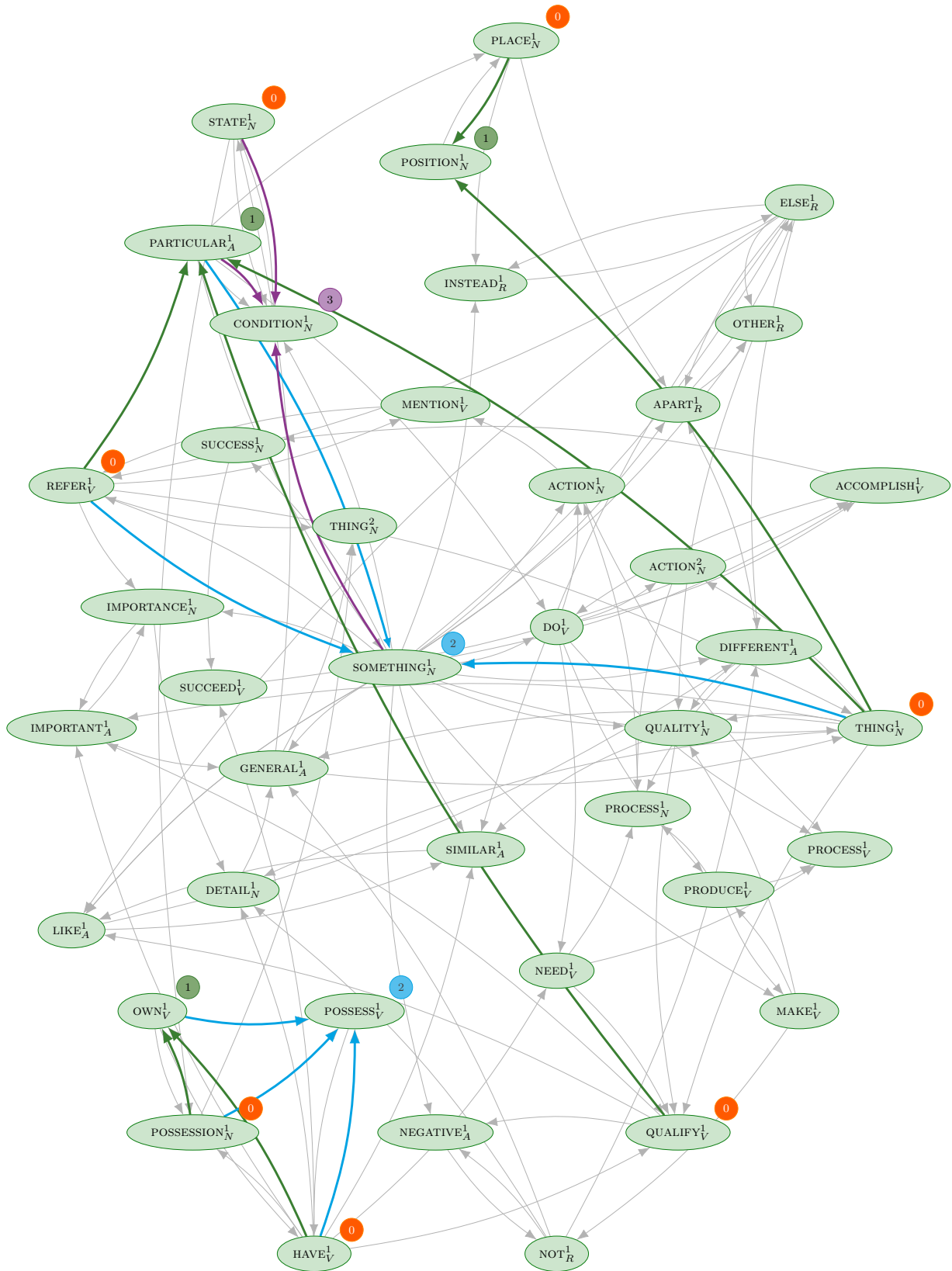


Figure 7. Graph associated to lexicon X_{large}
 (Lexemes are marked according to their k -reachability from U).

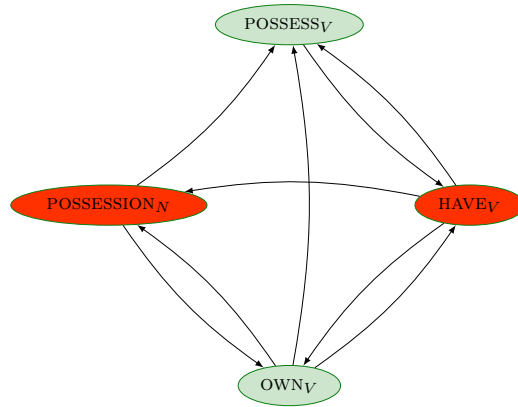


Figure 8. Graph associated to lexicon X_{small} (Lexemes in the minimum feedback vertex set are marked in red).

We assume that this form of learning is much less expensive than direct learning. It does not require the intervention of people, nor the participation of a third party to provide explanations; we remain in the exclusive sphere of words and meanings.

Nevertheless, in order to avoid falling into the trap of symbol grounding, we need to assume that a lexeme can only be learned by definition if it is completely defined, that is all lexemes in its definition are already known.

In our model, we use a monolingual lexicon as our external data source.

Within this model, our learning objective is stated as follows. Starting from an initial situation where we do not know the meaning of any lexeme, we aim to learn the meaning of all the lexemes of a lexicon. To do this, the learning process involves learning the lexemes one by one according to the following rules:

- 1) If a lexeme in the lexicon is unknown, but all the elements in its definition are known already, we learn it *by definition*.
- 2) Otherwise, we learn *directly* the next lexeme indicated by the learning strategy.
- 3) Repeat the previous steps until the entire lexicon is learned.

Learning Strategy:

Let us now formally define a learning strategy.

Definition 9 (Learning Strategy). Let $X = (\mathcal{A}, \mathcal{P}, \mathcal{L}, \mathcal{D})$ be a complete lexicon.

- (i) A *learning strategy* S is an ordered sequence of elements from \mathcal{L} .
- (ii) If the sequence S viewed as a set is a grounding set of X , we say that S is *exhaustive*.
- (iii) Otherwise, we say that S is *non exhaustive*.

In other words, a learning strategy for a lexicon is simply a list of lexemes in that lexicon sorted in the order in which they are to be learned. As we will see in the next section, this list can be derived from an external word list, for example

the Brysbaert and New [66] usage frequency list, or it can be determined using an algorithm. It is an exhaustive strategy if it allows us to learn all the lexemes in the lexicon.

Taking into account the two learning ways described above, we intuitively find that the learning effort for a strategy will be minimized if it requires to learn directly as few lexemes as possible. Without loss of generality, we further assume the cost of learning directly a lexeme to be 1 and 0 for learning by definition. We also say that a given strategy S_1 is more efficient than strategy S_2 if S_1 allows to fully learn the lexicon at a lower cost than S_2 .

Learning Algorithms:

We now expose the 3 algorithms that will let us calculate the cost of a learning strategy and determine if it is exhaustive.

Algorithm 1: Partial learning cost

The PARTIALCOST function computes the actual cost attributed to a strategy. As we mentioned before, some strategies, deemed non-exhaustive, fail to completely learn a lexicon. If so, PARTIALCOST calculates the cost so far and returns the portion of the lexicon that could not be learned. Otherwise, if the strategy is exhaustive, the cost returned corresponds to the total cost and there are no more lexeme to learn. Incidentally, this also allows us to verify if a strategy is exhaustive or not.

The function PARTIALCOST() accepts the following parameters:

- S , a learning strategy.
- X , a lexicon.

It returns as result the couple $(cost, X')$, where:

- $cost$ is the cost incurred by the strategy S for learning lexicon X ,
- X' is the remaining portion of X that could not be learned with S . X' can be used to determine if S is exhaustive:
 - If lexicon X' is empty, then the strategy S is exhaustive and $cost$ is equivalent to the total cost.
 - If lexicon X' is not empty, then strategy S is non-exhaustive. We must then use a fallback strategy to completely learn the lexicon and get the total cost.

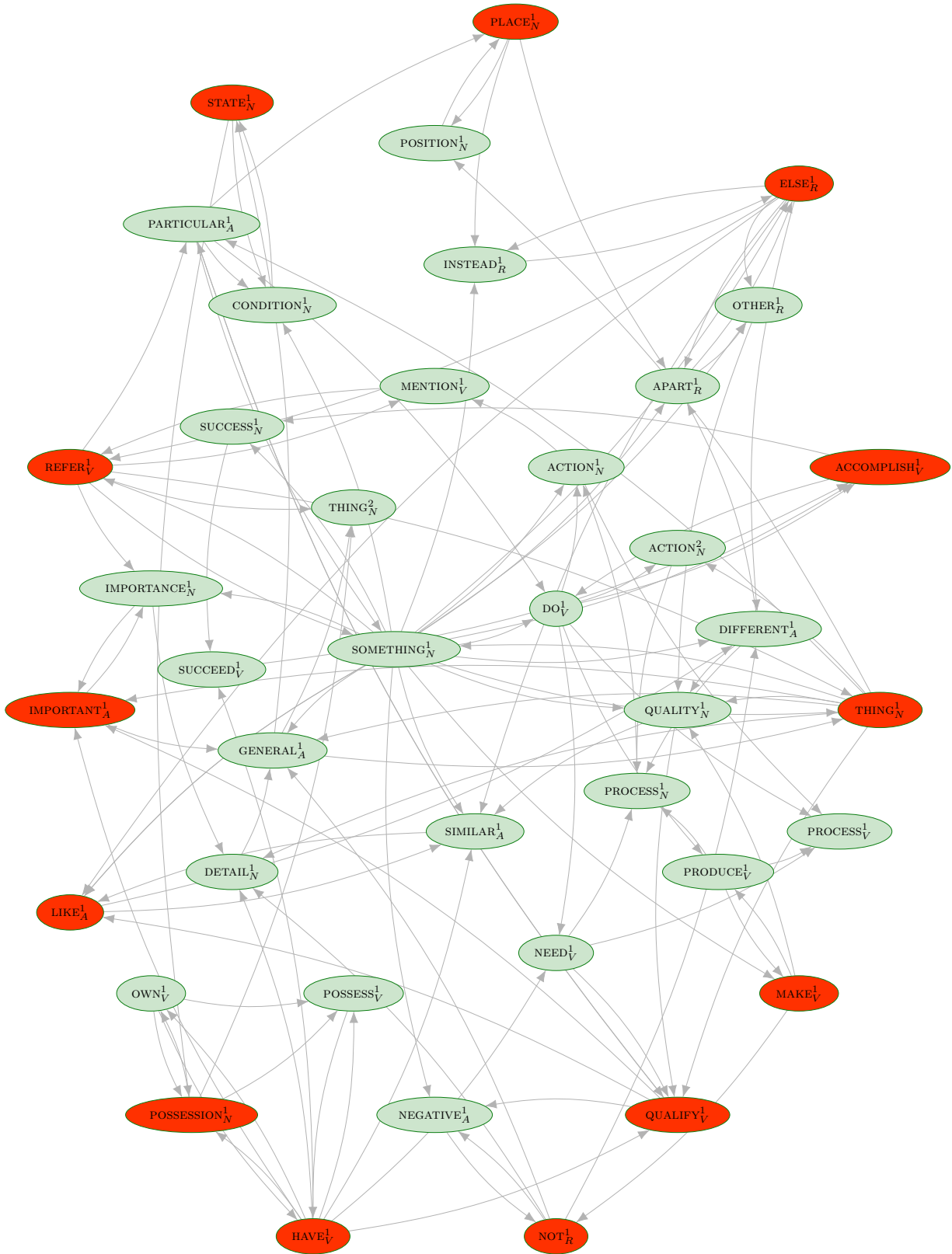


Figure 9. Graph associated to lexicon X_{large}
 (Lexemes in the minimum feedback vertex set are marked in red).

The flow of Algorithm 1 is as follows:

- line 4:** Get next lexeme from strategy.
line 5: Make sure lexeme exists in lexicon.
line 7: Learn lexeme directly, at cost 1.
lines 8-10: Then learn by definition, at cost 0, all lexemes completely defined.
line 12: Repeat the preceding steps until both the lexicon and the strategy are empty.
line 13: The cost of the strategy is the sum of the learning cost for all lexemes learnt directly.

Algorithm 1

```

1: function PARTIALCOST( $S$  : strategy,  $X$  : lexicon)
   : (cost, lexicon)
2:    $cost \leftarrow 0$ 
3:   while  $S \neq \emptyset$  and  $X \neq \emptyset$  do
4:      $\ell \leftarrow S.POP()$ 
5:     if  $\ell \in X$  then
6:       Remove  $\ell$  from  $X$ 
7:        $cost \leftarrow cost + 1$ 
8:       while  $\exists \ell' \in X$  with  $\deg^-(\ell') = 0$  do
9:         Remove  $\ell'$  from  $X$ 
10:      end while
11:     end if
12:   end while
13:   return ( $cost, X$ )
14: end function

```

Algorithm 2: Dynamic Degree Learning Cost

The DYNAMICCOST algorithm calculates the cost to learn all the lexemes in a lexicon. At each iteration, it chooses the node having the maximum out-degree amongst the ones remaining in the graph. In other words, it directly learns the lexeme appearing in the largest number of definitions. The function DYNAMICCOST() accepts only one parameter:

- X , a lexicon.

The flow of Algorithm 2 is:

- line 3:** Get the lexeme that corresponds to the highest out-degree node from associated graph.
line 6: Learn lexeme directly, at cost 1.
lines 7-9: Learn by definition, at no cost, all lexemes completely defined.
line 10: Get next lexeme with highest out-degree.
line 11: Repeat preceding steps until the lexicon and the strategy are empty.
line 12: The strategy's cost is the sum of the learning cost for all lexemes learnt directly.

Algorithm 2

```

1: function DYNAMICCOST( $X$  : lexicon) : cost
2:    $cost \leftarrow 0$ 
3:    $\ell \leftarrow$  lexeme whose out-degree is highest
4:   while  $\ell \neq \emptyset$  do
5:     Remove  $\ell$  from  $X$ 
6:      $cost \leftarrow cost + 1$ 
7:     while  $\exists \ell' \in X$  with  $\deg^-(\ell') = 0$  do
8:       Remove  $\ell'$  from  $X$ 
9:     end while
10:     $\ell \leftarrow$  lexeme whose out-degree is highest
11:  end while
12:  return  $cost$ 
13: end function

```

Algorithm 3: Total Learning Cost

This algorithm calculates the total cost incurred for learning all lexemes in lexicon X using strategy S .

For exhaustive strategies, the total cost obtained with this algorithm is identical to the one obtained with Algorithm 1. For non-exhaustive strategies, the total cost obtained is the sum of strategy S 's cost, plus the cost incurred by applying to the remaining portion X' of the lexicon a fallback strategy. It is theoretically possible to devise different algorithms that could be used as a fallback strategy. In our case, we use the DYNAMICCOST dynamic out-degree computation method described in Algorithm 2.

The parameters for the function TOTALCOST() are:

- S , a learning strategy.
- X , a lexicon.

It returns as result:

- *total cost*: the total cost incurred by learning all lexemes in X .

Algorithm 3

```

1: function TOTALCOST( $S$  : strategy,  $X$  : lexicon)
   : cost
2:   ( $cost, X'$ )  $\leftarrow$  PARTIALCOST( $S, X$ )
3:   ( $total\ cost$ )  $\leftarrow cost +$  DYNAMICCOST( $X'$ )
4:   return  $total\ cost$ 
5: end function

```

Complexity Analysis:

The algorithms described in the previous section are very efficient and easy to implement. Here is an evaluation of their complexity:

Algorithm 1 : If we take as hypotheses that:

- G is the graph associated with lexicon X .
- n is the number of vertices in G .
- m is the number of arcs in G .
- At line 6, vertex removal is done in $\mathcal{O}(1)$.
- At line 8, we only look at the neighbors of the deleted vertices.

Then, the time complexity is $\mathcal{O}(n + m)$ and the space complexity is $\mathcal{O}(n)$.

Algorithm 2 : If we take as hypotheses that:

- (a) G is the graph associated with lexicon X .
- (b) n is the number of vertices in G .
- (c) m is the number of arcs in G .
- (d) At line 5, vertex removal is done in $\mathcal{O}(1)$.
- (e) At line 7, we only look at the neighbors of the deleted vertices.
- (f) At lines 3 and 10 the list of candidates is managed using a priority queue, in time $\mathcal{O}(\log n)$.
- (g) The priority queue is implemented using a heap

The total cost for line 3 is $\mathcal{O}(m \log n)$, since each vertex v is processed in $\mathcal{O}(\log n)$ at most $\mathcal{O}(\deg(v))$ times. The time complexity is therefore $\mathcal{O}(m \log n)$ and the space complexity $\mathcal{O}(n)$.

Algorithm 3 : The time complexity for Algorithm 3 is therefore $\mathcal{O}(m \log n)$ and the space complexity is $\mathcal{O}(n)$.

IV. DATA SETS

In this section, we present the data we used to study of the structure of the dictionaries. First of all, we describe the digital dictionaries from which the lexicons and associated graphs were built. All of them are works from professional lexicographers and are published in electronic format. Then, we look at the different learning strategies developed to “learn” the words in the lexicons. They are of two types:

- *psycholinguistic strategies*, built from specially labeled word lists, called psycholinguistic norms,
- *algorithmic strategies*, obtained by analyzing the structure of graphs associated with the lexicons.

A. Digital Dictionaries

As a basis for the analysis of lexicon’s structure, we used eight different monolingual English-language dictionaries developed by professional linguists. Most of them are available in digital or paper format, with the exception of Wordsmyth, available only on the web.

The Cambridge International Dictionary of English (CIDE) is an English-language dictionary developed for ESL – English as a Second Language – students [7]. The version we used comprises about 19,000 articles and 47,000 lexemes.

The Longman Dictionary of Contemporary English (LDOCE) is an advanced dictionary also for ESL students. It was first published in 1978 [6]. It includes about 29,000 articles and 70,000 lexemes.

These 2 dictionaries, CIDE and LDOCE, have a common feature [67], [68]. They are both “monolingual learners dictionaries” (MLD), that is dictionaries developed especially for the needs of second language students, in this case English [69, p. 739, Rundell]. Both of them were built from their own control vocabulary. In other words, all definitions use only words from a restricted vocabulary, making it easier for novice users to understand definitions. In both cases, the control vocabulary contains about 2000 lexemes.

The Merriam-Webster’s Collegiate Dictionary (MWC) is the largest dictionary we studied [21]. The 11th edition includes more than 250,000 lexemes, grouped into 70,000 articles.

WordNet (WN) is not a dictionary in the true sense of the word. It is rather a lexical database of the English-language [69, p. 665, Fellbaum]. The different lexemes are

regrouped into synonym sets or **synsets**. Each synset then refers to a “meaning” and to a gloss – definition –. Synsets are also connected to each other by different types of semantic relations, such as hyponymy, hyperonymy, etc. The version we used, WordNet 3.0, contains about 132,000 lexemes grouped into 57,000 synsets.

According to its authors, Wordsmyth is at the same time a dictionary and a thesaurus [41]. Unlike CIDE and LDOCE, it does not use a control vocabulary. However it offers, in addition to the definition of a given word, information about its synonyms, antonyms and similar words [41]. It is available in 4 versions:

- The “Wordsmyth Educational Dictionary-Thesaurus” (WEDT) is the most comprehensive, comprising 73,000 lexemes. It was first developed in the 1980s.
- The “Wordsmyth Illustrated Learner’s Dictionary” (WILD) is an illustrated dictionary for children. It includes 4,200 lexemes.
- The “Wordsmyth Learner’s Dictionary-Thesaurus” (WLDT) is an intermediate level dictionary. It comprises 6,000 lexemes.
- The “Wordsmyth Children’s Dictionary-Thesaurus” (WCDT) is a beginners dictionary. It contains 20,000 lexemes.

Using a sequence of pre-treatments, we transformed all these digital dictionaries into disambiguated and complete lexicons. To do this, we first extracted from each dictionary the words with the desired parts of the speech: noun, verb, adjective and adverb. In addition, we did not considered the compound lexical items in our analysis; they were ignored during the transformation of dictionaries into lexeme graphs. We then lemmatized and pos-tagged the lexemes in the definitions with the “Stanford POS-tagger” [70], again ignoring the stop words. Finally, we disambiguated the lexemes using the first sense heuristic.

Table V presents some basic statistical data for the 8 lexicons considered:

- The number of lexemes in each dictionary (Lexemes).
- The number of lemmas (Lemmas).
- The average polysemy, being the average number of lexemes per lemma.
- The number of lexemes used in the definitions (Lexemes used).
- The ratio of the number of lexemes used vs the total number of lexemes (Usage Ratio).

TABLE V. Basic statistical data on lexicons

Lexicon	Lexemes	Lemmas	Polysemy	Lexemes Used	Usage Ratio
WILD	4 244	3 081	1.377	2 995	0.972
WLDT	6 036	3 433	1.758	2 212	0.644
WCDT	20 128	9 303	2.164	6 597	0.709
CIDE	47 092	18 694	2.519	8 773	0.469
LDOCE	69 204	22 511	3.074	10 074	0.448
WEDT	73 091	28 986	2.522	18 197	0.628
WN	132 547	57 243	2.316	29 600	0.517
MWC	249 137	68 181	3.654	33 533	0.492

After building the graphs associated with the lexicons, we then analyzed their structure. Many measures can be applied

to networks or graphs. Among others, Batagelj *et al.*, identify a series of measures specifically aimed at dictionary graphs [71]. For our analysis, we selected the numbers that present a quick overview of the graphs. Table VI shows the results obtained from the graphs associated with the 8 lexicons:

- The number of vertices (Nodes).
- The number of arcs (Arcs).
- The number of strongly related components (SCCs).
- The number of lexemes in the largest SCC (<SCC).
- The diameter of the largest SCC (Diameter), being “[...] the largest number of vertices that must be traversed in order to travel from one vertex to another” [72,].
- The density of the graph (Density).
The density of a graph $G = (V, E)$ is the ratio of the number of arcs $|E|$ in G over the maximum number of arcs possible $= (|V| \cdot (|V| - 1)) / 2$ [73].
- The Characteristic Path Length (CPL) – the average length of the shortest paths – is calculated for a graph $G = (V, E)$ using the following formula [74]:

$$\sum_{u,v \in V} \frac{d(u,v)}{|V|(|V| - 1)}$$

TABLE VI. Associated graphs structural data

Lexicon	Nodes	Arcs	SCCs	<SCC	Diam.	Dens.	CPL
WILD	4 244	45 789	2 750	1 446	17	10.79	1.75
WLDT	6 036	28 623	5 088	858	25	4.74	1.10
WCDT	20 128	102 657	17 551	2 341	22	5.10	0.87
CIDE	47 092	334 888	45 306	1 702	16	7.11	0.21
LDOCE	69 204	415 052	67 224	1 770	16	6.00	0.16
WEDT	73 091	362 569	67 318	5 056	29	4.96	0.61
WN	132 547	694 067	124 589	7 079	30	5.24	0.50
MWC	249 137	1 155 085	239 478	8 842	29	4.64	0.31

B. Learning Strategies

There are a very large number of different strategies for learning all the words of a dictionary or lexicon. One could imagine trying them all. If a lexicon contains n lexemes, there are then $n!$ different ways to order them to specify a learning order. Except for trivial cases, it is obviously impossible to evaluate all those possibilities. We decided to restrict our study to two kinds of strategies:

- *Psycholinguistic Strategies*: These strategies are based on lists of words ordered according to psycholinguistic properties.
- *Algorithmic Strategies*: These strategies are built using algorithms from graph theory. Among these, one can distinguish the adapted strategies, built solely for a specific lexicon, and the global strategies based on normalized structural properties common to all lexicons.

Psycholinguistic Strategies:

Researchers interested in the cognitive aspects of language have long used standardized databases, called *psycholinguistic norms*, which group words according to their psycholinguistic properties [75]–[78]. For example, the MRC database lists 150,837 English-Language words, for which 26 different psycholinguistic properties are listed [78].

Among the recent psycholinguistic norms, we have selected five of them, made available by their authors as a supplement to their research work. They are lists of words based on psycholinguistic variables frequently used by researchers in language psychology: words *usage frequency*, *age of acquisition* and *degree of concreteness* [79]. The *usage frequency* measurement is probably the most commonly used norm for psycholinguistic research [66]. It is a measure of the rate of occurrence of words within a given corpus, normalized to 1 million. *Age of acquisition* is an estimation of the age at which children are presumed, on average, to have learned a word. As for the *concreteness*, it “[...] refers to the degree to which words refer to individuals, places and objects that can be seen, heard, touched, smelled or tasted” [80, cited by [75]].

Table VII presents the 5 data sources we used to construct our learning psycholinguistic strategies, versus the psycholinguistic variables from which they were derived. It goes without saying however that our analysis could easily be extended to other databases or other variables, depending on data availability.

TABLE VII. Psycholinguistic Variables and Learning Strategies

Variable	Strategy	Source	# Words
Usage frequency	FREQ _{Brybaert}	[66]	74 000
Usage frequency	FREQ _{NGSL+}	[65], [81], [82], [83]	6 600
Age of acquisition	AOA _{Brybaert}	[84]	31 000
Age of acquisition	AOA _{Childes}	[85]	13 000
Concreteness	CONC _{Brybaert}	[86]	37 000

To build our learning strategies, we first lemmatized and disambiguated the words from the databases in order to transform them into lexemes, and then ordered them according to the psycholinguistic variable considered. For example, for a strategy based on the age of acquisition, the first lexeme proposed by the strategy corresponds to the word that the authors consider to be learned the earliest in the development of the child. Then the second lexeme suggested corresponds to the second word learned and so on until we get to the lexeme estimated to be learned the latest.

An additional alignment step between lexicons and strategies is required. Since the psycholinguistic data used to construct the strategies come from heterogeneous sources, the lexemes they contain do not necessarily match with the lexicons. When a lexeme proposed by a strategy does not appear in a lexicon, we choose to simply ignore it. In particular, we do not measure the degree alignment of psycholinguistic strategies with lexicons, that is, the size of the intersections between the strategies and the lexicons. This is one of the limitations of our analysis. If we were to tackle it in the future, this could possibly allow a more refined assessment of the quality of the strategies.

Let us now look at how the different learning strategies were developed.

The first strategy in table VII, FREQ_{Brybaert} is derived from the norm described in [66]. The authors assembled it from SUBTLEX_{US}, a corpus of film subtitles in American English. It includes 74,000 unlemmatized words.

The FREQ_{NGSL+} strategy comes from lists of words used to learn English as a second language. Although word lists

are not based solely on psycholinguistic criteria, they are still an important domain of research since the works of Ogden [59] and West [61]. For this paper, we selected the “New General Service List” (NGSL) from Browne, Culligan and Phillips [65]. This is an improved version of West’s original list, containing 2,800 words selected from the Cambridge English Corpus (CEC). To enable the NGSL to be more easily compared to other psycholinguistic strategies, we developed an augmented version: the NGSL+. The latter is obtained by concatenating to the NGSL three other lists of complementary words developed by the authors of the NGSL from specialized corpora:

- The New Academic Word List (NAWL) is constructed from a body of academic texts [81]. It contains 963 words.
- The Business Service List (BSL) is a list of 1700 words related to business and commerce [82].
- The “TOEIC Service List” (TSL) is intended for students wishing to attain the “Test of English for International Communication” (TOEIC) certification. It is a list of 1200 words that complement the NGSL [83].

To build the $AOA_{Brysbaert}$ strategy, we used the norm based on the age acquisition norm from Kuperman *et al.* [84]. Since it is not possible to get this information directly from the children themselves, the most frequently used method is to interview adults and ask them to assess the age at which they have learned certain words. For their research, Kuperman, Stadthagen-Gonzalez and Brysbaert used a crowdsourcing technique based on the Amazon Mechanical Turk. Adult participants were asked to estimate how old they were when they learned the words from a list. From their responses, the authors constructed a list of 31,000 words tagged with their estimated age of acquisition, ranging from 1 to 21 years old.

The other age-based acquisition strategy, $AOA_{Childes}$, uses data from another source: the project “Child Language Data Exchange System” (CHILDES) [85]. In this case, a different method was used to collect the data. The age of acquisition was estimated from recorded conversations of children aged 1 to 11 years. The resulting list, noisier than the previous one, contains 13,000 words.

For the $CONC_{Brysbaert}$ strategy, we used Brysbaert, Wariner and Kuperman’s norm [86]. As with their study on the age of acquisition, the authors used crowdsourcing to recruit participants. The adults chosen had to classify words on a concreteness scale ranging from 1 to 5, 1 being completely abstract and 5 corresponding to the most concrete words. For example, the concrete words *banana*, *apple* and *baby* are of degree 5, while *belief* and *although* are respectively of degree 1.19 and 1.07. The list thus created contains 37,000 words.

Algorithmic Strategies:

Algorithmic strategies are lists of lexemes derived from the structural properties of graphs, which means that lexemes are ordered according to the results of graph theory algorithms.

Table VIII summarizes the algorithmic strategies we have experimented with. It should be noted that all these strategies directly use the *COST* or *DYNAMICCOST* algorithms without resorting to a fallback strategy. In contrast to psycholinguistic strategies, the techniques used ensure that lexicons are fully “learned” when the algorithms terminate.

With the first 3 strategies, $MFVS_{<lex>}$, $DD_{<lex>}$ and $SD_{<lex>}$, we get as many different strategies as lexicons,

TABLE VIII. Algorithmic learning strategies

Strategy	Property	Algorithm	Number
$MFVS_{<lex>}$	Min. Grounding Set	<i>COST</i>	8 (1 per lexicon)
$DD_{<lex>}$	Dynamic Degree	<i>DYNAMICCOST</i>	8 (1 per lexicon)
$SD_{<lex>}$	Static Degree	<i>COST</i>	8 (1 per lexicon)
$MFVS_{mixed}$	Min. Grounding Set	<i>COST</i>	1
DD_{mixed}	Dynamic Degree	<i>COST</i>	1
SD_{mixed}	Static Degree	<i>COST</i>	1

each one of them being adapted to a specific lexicon. Here the $<lex>$ index represents the lexicon. For instance, SD_{LDOCE} corresponds to the static degree strategy for the LDOCE lexicon.

The $MFVS_{<lex>}$ strategies are assembled individually for each lexicon $<lex>$ from the minimum grounding set calculated with the method described in definition 8. Although the problem of calculating a *MFVS* is NP-hard in general, it was still possible to obtain an optimal solution for 6 of the 8 lexicons and a good approximation for the 2 others. In this specific case, the order of the lexemes in the strategy is not considered.

With the $DD_{<lex>}$ Dynamic Degree strategies, the next lexeme to be learned is not chosen from a predetermined list. As described in Algorithm 2, it is calculated dynamically at each step by selecting the vertex whose out-degree is the highest. Since “learned” lexemes are systematically removed at each step, it is equivalent to selecting each time the lexeme that appears in the greatest number of definitions.

For the $SD_{<lex>}$ Static Degree strategies, the next lexeme to learn comes from a list containing all the lexicons of the lexicon. The lexemes are ordered in descending order of the out-degree of their corresponding vertices. Unlike the $DD_{<lex>}$ strategies, the degree of vertices is computed statically when the graph is initially built. Thus, one begins to learn the lexemes from the one that is used in most definitions, going to the least used.

In order to evaluate whether the use of strategies uniquely built for each lexicon could distort the results, we also developed global strategies, based on structural data common to all the lexicons. Those strategies, called *mixed strategies*, are assembled by merging into one global list all the lexemes coming from the strategies adapted to each lexicon. For example, the lexemes from the 8 $DD_{<lex>}$ strategies are merged to form the DD_{mixed} list. It is built by randomly choosing one of the 8 lexicons, and then selecting the next lexeme from the corresponding strategy. If the lexeme is already in the global list, it is ignored. We then repeat this process until all the lists are exhausted. For example, the DD_{mixed} strategy was built by concatenating the lexemes in the order shown in Table IX.

V. RESULTS AND DISCUSSION

In this section, we present the results obtained during our experiments. First, we explain the different measures collected during the execution of the algorithms on the lexicons. We then show comparative results for the various learning strategies and the 8 lexicons analyzed. We conclude the section with a discussion of the results.

TABLE IX. Mixed learning strategies

Number	Lexeme	Origin
1.	BE;V	DD _{WILD}
2.	HAVE;V	DD _{WN}
3.	PERSON;N	DD _{WLDT}
4.	USE;N	DD _{WN}
...
5990.	DEALFISH;N	DD _{MWC}
5991.	PHENYTOIN;N	DD _{MWC}

A. Measurements

To allow for the combinations of strategies and lexicons to be compared, different performance indicators were recorded during the tests.

Detailed Learning Measurements:

When learning a lexicon with a strategy, a series of values is recorded each time a lexeme is learned directly. This makes it possible to evaluate the pace of the learning process. Table X shows an overview of the data recorded during one learning cycle of the MWC lexicon using the AOA_{Brybaert} strategy.

TABLE X. Learning progress (partial)

Cost	Nodes	Arcs	Degree	Lexeme	Fallb.
1	249 056	1 152 896	2	mama;n	0
2	249 054	1 152 894	2	mom;n	0
3	249 053	1 152 892	8	potty;n	0
4	249 051	1 152 884	17	yes;n	0
5	249 047	1 152 867	1 522	water;n	0
6	249 039	1 151 337	130	wet;a	0
7	249 037	1 151 208	33	spoon;n	0
8	249 036	1 151 175	51	nap;n	0
9	249 030	1 151 121	2	daddy;n	0
10	249 028	1 151 119	18	hug;n	0
11	249 026	1 151 101	212	shoe;n	0
10	249 028	1 151 119	18	hug;n	0
11	249 026	1 151 101	212	shoe;n	0
...
10113	14	14	1	kakemono;n	484
10114	12	12	1	stillbestrol;n	485
10115	10	10	1	ciphertext;n	486
10116	8	8	1	banderilla;n	487
10117	6	6	1	amphitryon;n	488
10118	4	4	1	mannose;n	489
10119	2	2	1	phenytoin;n	490

We can see the following measures:

Cost: Number of lexemes learned directly since the beginning of the cycle

Nodes: Number of vertices remaining in the graph (before learning the lexeme)

Arcs: Number of arcs remaining in the graph (before learning the lexeme)

Degree: Out-degree of the lexeme

Lexeme: Lexeme learned

Fallb.: Cumulative cost of the fallback strategy

Global Performance Measurements:

At the end of a learning cycle, global performance indicators are also recorded. Table XI shows, for each combination of lexicons and learning strategies, their performance in terms of cost, efficiency, percentage of words learned directly, and coverage (if applicable).

The counters shown are:

Cost: Indicates the total learning cost for the strategy, i.e., the total number of lexemes that had to be learned directly in order to successfully learn the full lexicon (see algorithms 1 and 2). For example, the cost for the DD_{WILD} strategy and the WILD lexicon is 574. This represents the total number of lexemes that had to be learned directly.

Efficiency: Efficiency is the ratio of the total number of lexemes over the cost of learning. We see that the WILD lexicon contains 4,244 lexemes and that 1,260 lexemes had to be learned directly with the FREQ_{NGSL+} strategy. The efficiency of the FREQ_{NGSL+} for the WILD lexicon is therefore 4,244/1,260 or 3.37. We can interpret this measure as the average number of lexemes that can be learned by definition for each lexeme learned directly. In other words, we learn on average 2.37 additional lexemes by definition every time we learn a lexeme directly.

Pct: Percentage of words learned directly for a given strategy and lexicon. This corresponds to the proportion of the number of lexemes learned directly, relative to the total number of lexemes in the lexicon. For example, for the WEDT lexicon and the FREQ_{NGSL+} strategy, the percentage of lexemes learned directly is 3 238 over 73 091 or 4.43%.

Coverage: Valid for non-exhaustive strategies only, this number measures the efficiency of the strategy, as a percentage of the total cost, vs the fallback strategy. For example, for the WCDT lexicon and the FREQ_{NGSL+} strategy, the coverage is 82.9%. This means that out of a total learning cost of 1,354, 1,122 lexemes, or 82.9%, were learned with the FREQ_{NGSL+} strategy. The remaining 232 (17.1%) were learned with DD, our fallback strategy. On the other hand, for this same WCDT lexicon and the FREQ_{Brybaert} strategy, the coverage reaches 99.8%.

It turns out that, unsurprisingly, the most efficient strategies are those that take advantage of the minimal grounding set: the MFVS_{<lex>}. Coming right after, the strategies optimized according to the vertices out-degree - DD_{<lex>} and DS_{<lex>} - are also very efficient. We also remark that for some lexicons - MWC, WN, WEDT, WCDT - the FREQ_{NGSL+} and AOA_{Childes} strategies have a low coverage rate of less than 90%.

B. Discussion

Global Performance Measurements:

The Figures in this section (Best viewed in colors) compare different aspects of the learning process for the 8 lexicons studied. Each of the sub-figures is produced using the detailed performance measurements recorded while lexemes are being learned.

The first series of graphs in Figure 10 compare the learning rate of algorithmic strategies versus psycholinguistic strategies.

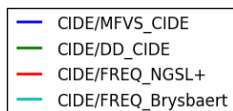
To facilitate comprehension, let us examine Figure 10a for the CIDE lexicon. It shows the learning rate for the algorithmic strategies MFVS_{CIDE} and DD_{CIDE} in comparison with the FREQ_{NGSL+} and FREQ_{Brybaert} psycholinguistic strategies.

TABLE XI. Cost, efficiency, percentage and coverage

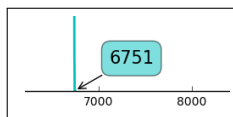
Strategy	Measure # Lex	CIDE	LDOCE	MWC	WN	WEDT	WCDT	WLDT	WILD
		47 092	69 204	249 137	132 547	73 091	20 128	6 036	4 244
MFVS	Cost	349	484	1 544	1 251	1 365	570	231	340
	Eff.	134.93	142.98	161.36	105.95	53.55	35.31	26.13	12.48
	Pct	0,74%	0,70%	0,62%	0,94%	1,87%	2,83%	3,83%	8,01%
	Cov.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.
DD	Cost	684	843	3 095	2 566	2 389	897	394	574
	Eff.	68.85	82.09	80.50	51.66	30.59	22.44	15.32	7.39
	Pct	1,45%	1,22%	1,24%	1,94%	3,27%	4,46%	6,53%	13,52%
	Cov.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.
DS	Cost	687	838	3 081	2 558	2 386	899	394	577
	Eff.	68.55	82.58	80.86	51.82	30.63	22.39	15.32	7.36
	Pct	1,46%	1,21%	1,24%	1,93%	3,26%	4,47%	6,53%	13,60%
	Cov.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.
MFVS _{mixed}	Cost	704	966	3 077	2 835	2 348	957	398	612
	Eff.	66.85	71.64	80.96	46.75	31.13	21.03	15.17	6.93
	Pct	1,49%	1,40%	1,24%	2,14%	3,21%	4,75%	6,59%	14,42%
	Cov.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.
DD _{mixed}	Cost	768	963	3 466	3 002	2 574	987	448	645
	Eff.	61.32	71.82	71.88	44.15	28.39	20.39	13.47	6.57
	Pct	1,63%	1,39%	1,39%	2,26%	3,52%	4,90%	7,42%	15,20%
	Cov.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.
DS _{mixed}	Cost	793	988	3 776	3 021	2 721	1 024	454	678
	Eff.	59.32	70.00	65.98	43.87	26.86	19.65	13.30	6.25
	Pct	1,68%	1,43%	1,52%	2,28%	3,72%	5,09%	7,52%	15,98%
	Cov.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.	s. o.
FREQ _{NGSL+}	Cost	2 813	1 954	5 008	4 127	3 238	1 354	712	1 260
	Eff.	16.74	35.42	49.75	32.12	22.57	14.87	8.48	3.37
	Pct	5,97%	2,82%	2,01%	3,11%	4,43%	6,73%	11,80%	29,69%
	Cov.	97.0%	90.4%	71.2%	73.4%	67.7%	82.9%	97.9%	92.8%
FREQ _{Brysb}	Cost	6 751	2 170	8 217	7 204	6 555	1 999	960	1 193
	Eff.	6.98	31.89	30.32	18.40	11.15	10.07	6.29	3.56
	Pct	14,34%	3,14%	3,30%	5,44%	8,97%	9,93%	15,90%	28,11%
	Cov.	99,9%	99,3%	96,1%	94,8%	98,7%	99,8%	99,7%	99,6%
AOA _{Chil}	Cost	4 971	5 010	7 729	7 284	5 586	3 409	1 585	2 016
	Eff.	9.47	13.81	32.23	18.20	13.08	5.90	3.81	2.11
	Pct	10,56%	7,24%	3,10%	5,50%	7,64%	16,94%	26,26%	47,50%
	Cov.	99,4%	97,7%	82,9%	86,3%	84,3%	97,3%	99,7%	98,3%
AOA _{Brysb}	Cost	7 105	4 851	10 119	10 340	8 278	2 950	1 284	1 430
	Eff.	6.63	14.27	24.62	12.82	8.83	6.82	4.70	2.97
	Pct	15,09%	7,01%	4,06%	7,80%	11,33%	14,66%	21,27%	33,69%
	Cov.	99,6%	99,2%	95,2%	94,0%	96,7%	97,6%	99,5%	95,7%
CONC _{Brysb}	Cost	8 900	11 669	16 580	17 037	12 792	6 042	2 373	2 477
	Eff.	5.29	5.93	15.03	7.78	5.71	3.33	2.54	1.71
	Pct	18,90%	16,86%	6,65%	12,85%	17,50%	30,02%	39,31%	58,36%
	Cov.	99,7%	99,6%	96,4%	96,0%	97,5%	98,9%	99,7%	97,6%

We can see:

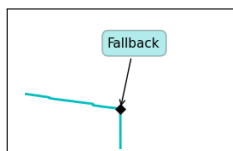
- The curves illustrating the learning rate, identified by a different color for each lexicon:



- A tile of the same color as the curve, showing for each strategy the total cost incurred:



- Another tile showing, for each non-exhaustive strategy, the point where it was required to resort to the fallback strategy:



For the CIDE lexicon (Figure 10a) as well as for all the lexicons in Figure 10, we see that the MFVS strategy is the most efficient one. This confirms the hypothesis that learning the minimal grounding set lexemes allows to quickly break the definition loops.

The graphs in Figure 11 show the learning rate for dynamic degree strategies versus those based on the static degree. We note that these two algorithmic strategies, DD <LEX> and SD <LEX> give in practice equivalent results.

The graphs in Figure 12 compare the learning rate for the algorithmic strategy DD<LEX> versus the psycholinguistic strategies FREQ_{NGSL+}, FREQ_{Brysb}, AOA_{Brysb} and CONC_{Brysb}. We see that psycholinguistic strategies are much less effective in breaking the definition loops. Since the lexemes order is decided according to psycholinguistic criteria, many lexemes are learned directly and increase the cost of the strategy, whereas they could have been learned by definition - at zero cost - later in the learning cycle. Among the psycholinguistic strategies, the two frequency-based strategies, FREQ_{NGSL+} and FREQ_{Brysb}, are the most effective, whereas the CONC_{Brysb} strategy is clearly less efficient. Intuitively, we see that it is not possible to succeed in learning all the

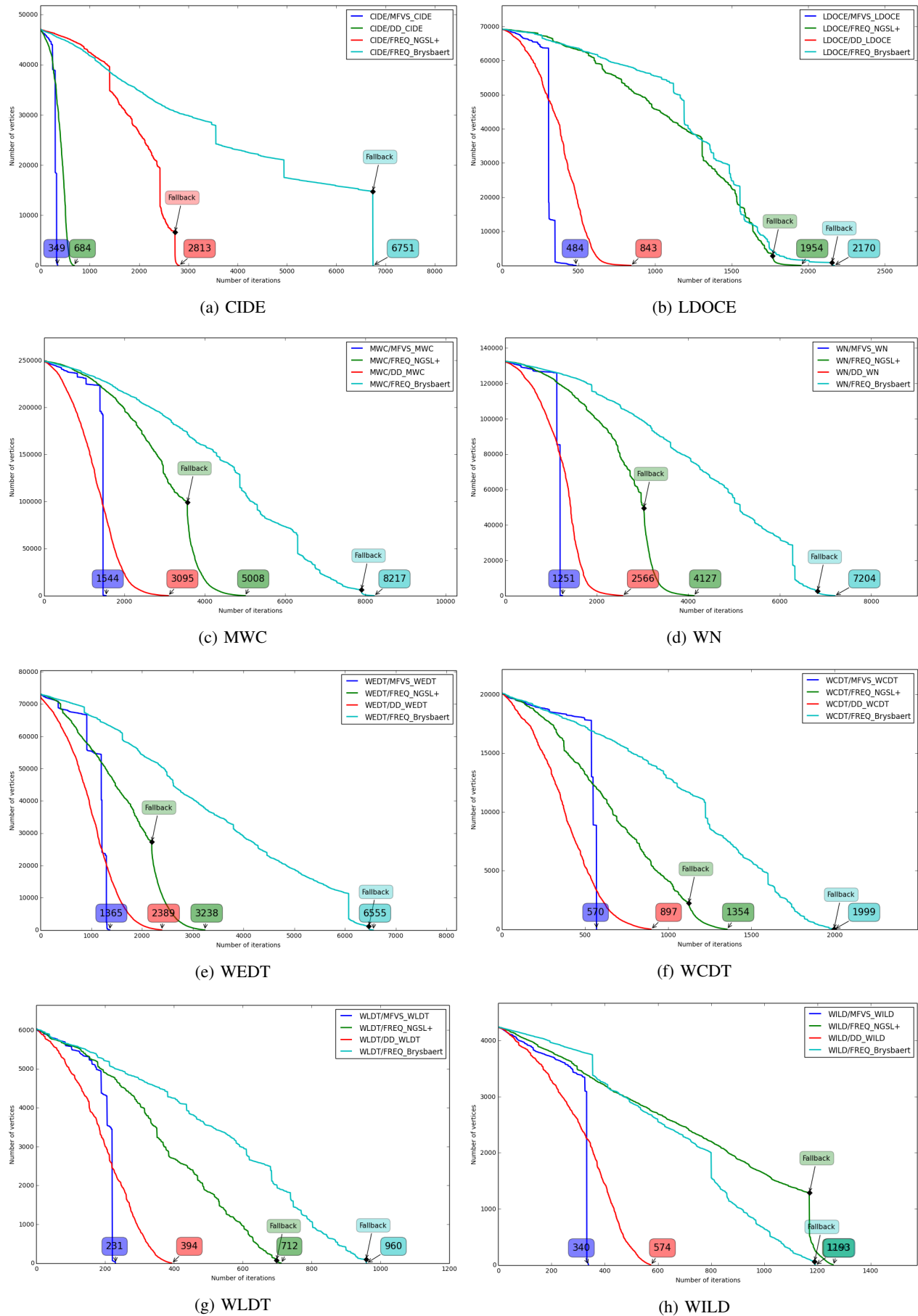


Figure 10. Learning: Algorithmic vs Psycholinguistic Strategies

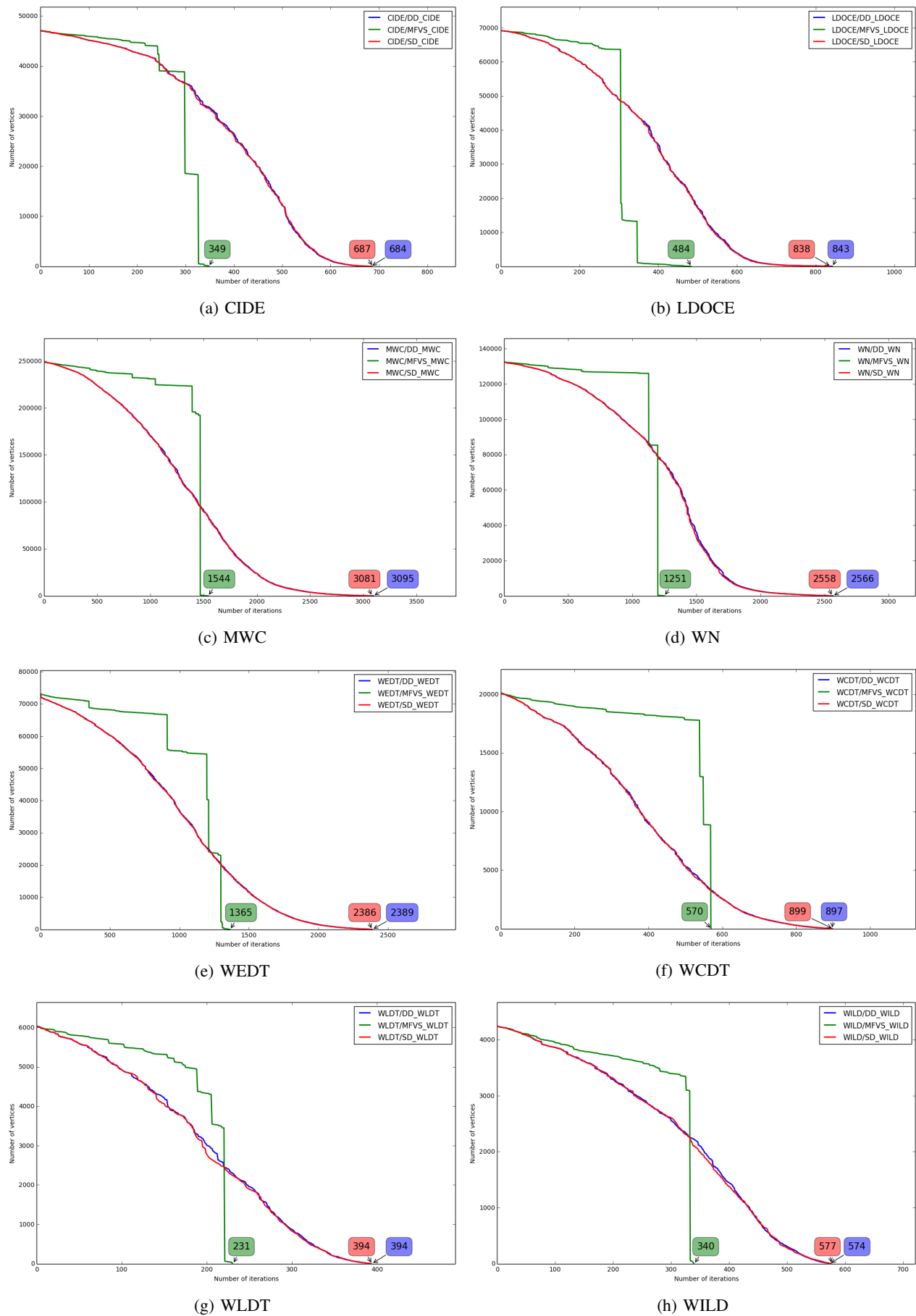


Figure 11. Learning: Dynamic vs Static Strategies

words of a dictionary by using only concrete words. One must combine two kinds of words, abstract and concrete, to build definitions that properly translate the lexical meaning.

Figure 13 compares the learning rate of age-based strategies. At first glance, AOA_{Childes} seems to offer a better efficiency than AOA_{Brybaert} . However, since OA_{Childes} contains far fewer lexemes than AOA_{Brybaert} , its coverage is lower. The fallback point is reached very soon, before having learned 40% of the lexemes. In this case, the use of a fallback strategy makes direct comparison between AOA_{Childes} and AOA_{Brybaert} difficult. That being said, although the AOA_{Childes} and AOA_{Brybaert} strategies are not the most successful ones, they show that in most cases it is sufficient to know less than 15% of the lexemes to learn the rest by definition.

The graphs in Figure 14 compare the mixed algorithms learning rate against the most effective psycholinguistic strategies. Since they are optimized to use as few lexemes as possible, algorithmic strategies are clearly more efficient. Lexemes ordering is the key factor making one strategy more efficient than the other. We notice large differences in this regard when comparing strategies based on the same psycholinguistic criteria. For example, the word dog appears at 485th rank in AOA_{Childes} , while it is ranked 25th in AOA_{Brybaert} .

Efficiency:

Table XI presents overall performance measurements for learning strategies. Figure 15 (best viewed in colors) plots efficiency for each evaluated lexicon and strategy. Each lexicon is each represented by a color coded curve. The strategies are shown on the X axis, from left to right in descending order of efficiency.

We can distinguish 3 different groups of strategies:

- 1) The 1st group comprises only one strategy: the algorithmic $MFVS_{\langle \text{lex} \rangle}$. For every lexicon, it is clearly the most efficient.
- 2) The second group gathers the other graph algorithmic strategies. $DD_{\langle \text{lex} \rangle}$ and $SD_{\langle \text{lex} \rangle}$ are uniquely optimized for each lexicon, while $MFVS_{\text{mixte}}$, DD_{mixte} and SD_{mixte} are global strategies common to all lexicons. They are less efficient than $MFVS_{\langle \text{lex} \rangle}$, but still very good.
- 3) Finally, the third group brings together the psycholinguistic strategies $FREQ_{\text{NGSL+}}$, $FREQ_{\text{Brybaert}}$, AOA_{Childes} , AOA_{Brybaert} and $CONC_{\text{Brybaert}}$. Their performance is clearly inferior compared to algorithmic strategies.

In summary, the uniquely optimized strategies, $MFVS_{\langle \text{lex} \rangle}$, $DD_{\langle \text{lex} \rangle}$ and $SD_{\langle \text{lex} \rangle}$, are the most efficient ones. As for the question of whether it is possible to develop “general” strategies as efficient as “lexicon specific” strategies, the mixed strategies show that this is possible. The 3 mixed strategies, $MFVS_{\text{mixte}}$, DD_{mixte} and SD_{mixte} are almost as efficient as the “lexicon specific” strategies. For each lexicon, they perform much better than strategies based on psycholinguistic variables.

VI. CONCLUDING REMARKS

By definition, a traditional dictionary is a closed world. According to Amsler, “[...] the dictionary is a closed system, i.e., words used in definitions are defined elsewhere in the dictionary” [87, p. vii]. It is therefore possible to build a graph structure from the words of a dictionary and the definitions

that link them together. In this article, our goal was to use graph theory and algorithms to study these dictionary graph structure.

Although the terms dictionary and word may seem a priori clear and unambiguous, their imprecision makes them unsuited for rigorous mathematical analysis. We decided to replace them in our discussion by more precise terms: lexicons and lexemes. We also established a linguistic terminology allowing to formally define the notions of lexicon and associated graph.

In order to explore the structure of lexicons, we have shown the interest of using a formal word learning process as an analytical tool. We considered that a word - a lexeme - can be learned in two ways: by definition, when all the lexemes in its definition are already known, and by direct learning, when one needs to invest significant effort to ground it through some sensorimotor perception. We also described our learning model, as well as related strategies and algorithms aiming to minimize the effort and cost required to learn all the lexicons of a lexicon.

Subsequently, we described the source data used to carry out our analyzes: monolingual digital dictionaries and psycholinguistic norms.

Finally we exposed our results in two different ways:

- in terms of learning rate, which is a measure of how quickly a strategy progresses toward its goal of learning all lexemes;
- in terms of efficiency, being the ratio between the number of lexemes learned by definition and the number of lexemes learned directly.

Our analysis confirmed the results of other researches ([15], [16], [55]). Circular relationships between words play a key role in the organization and structure of dictionaries.

If we consider a dictionary from the strict point of view of its utility for the reader, the definition of a word will be relevant insofar as the latter already knows all the words that make up this definition, or at least enough words to understand the intended meaning.

“The usefulness of a dictionary definition depends on its ability to explain a meaning using words the reader already knows” [88].

If this is not the case, the reader must look for unknown words. And in all dictionaries, there are necessarily many circular definitions:

“In a typical dictionary, more than a quarter of all definitions are written using words whose definitions ultimately refer back to the word being defined” [88]

A reader who does not know enough of the language will inevitably encounter intractable definition loops. Our analysis has shown that the most efficient learning strategies are those that break those definition loops as quickly as possible. In this regard, those who use the minimum grounding set - feedback vertex set of the associated graph - work best. However, the problem of finding a feedback vertex set is NP-hard. Even using advanced approximation techniques, this remains a complex calculation.

Our results show that alternative strategies, built using simple graph properties, can also be very efficient. For example, with a strategy ordering lexemes according to the out-degree

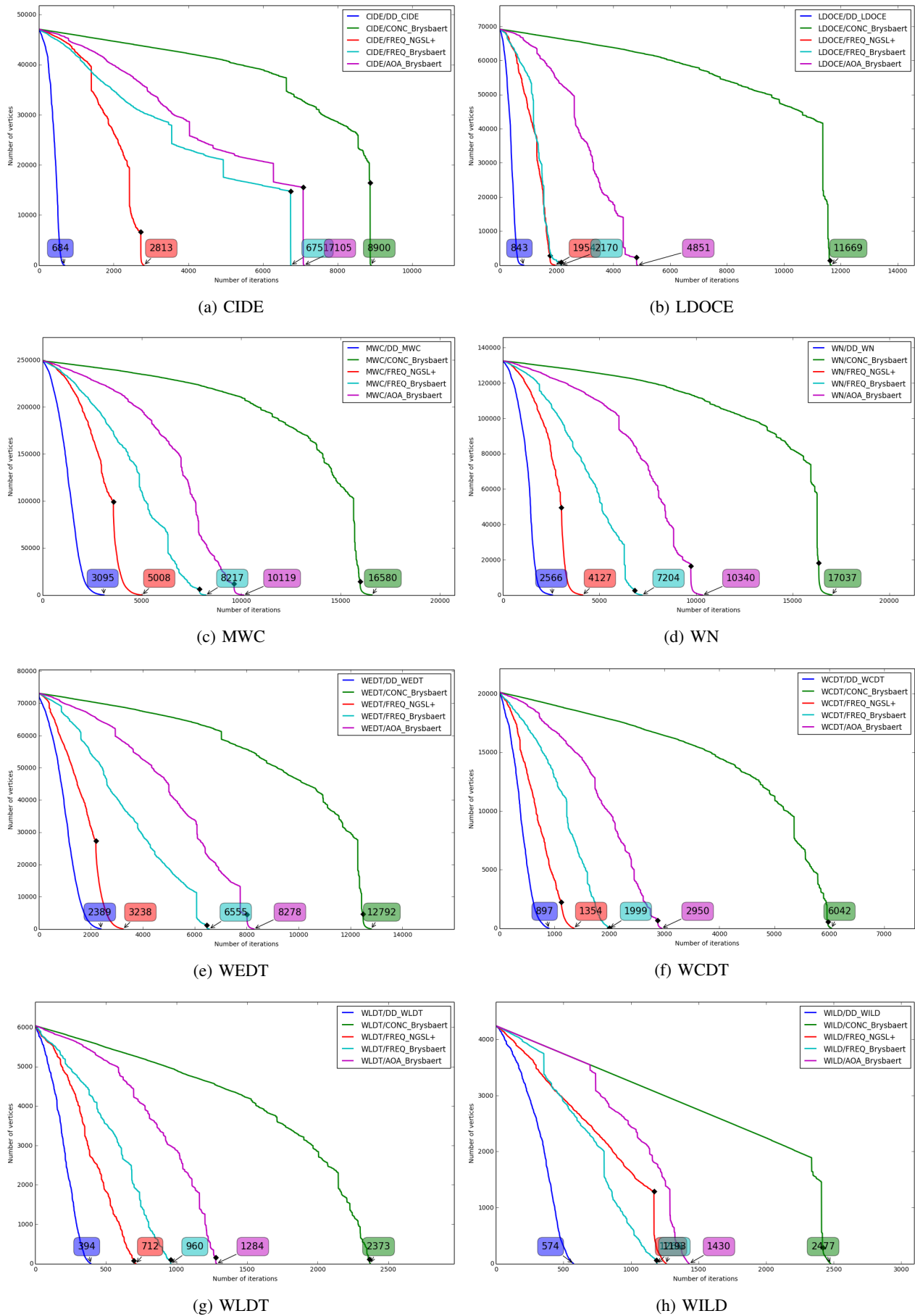


Figure 12. Learning: Frequency

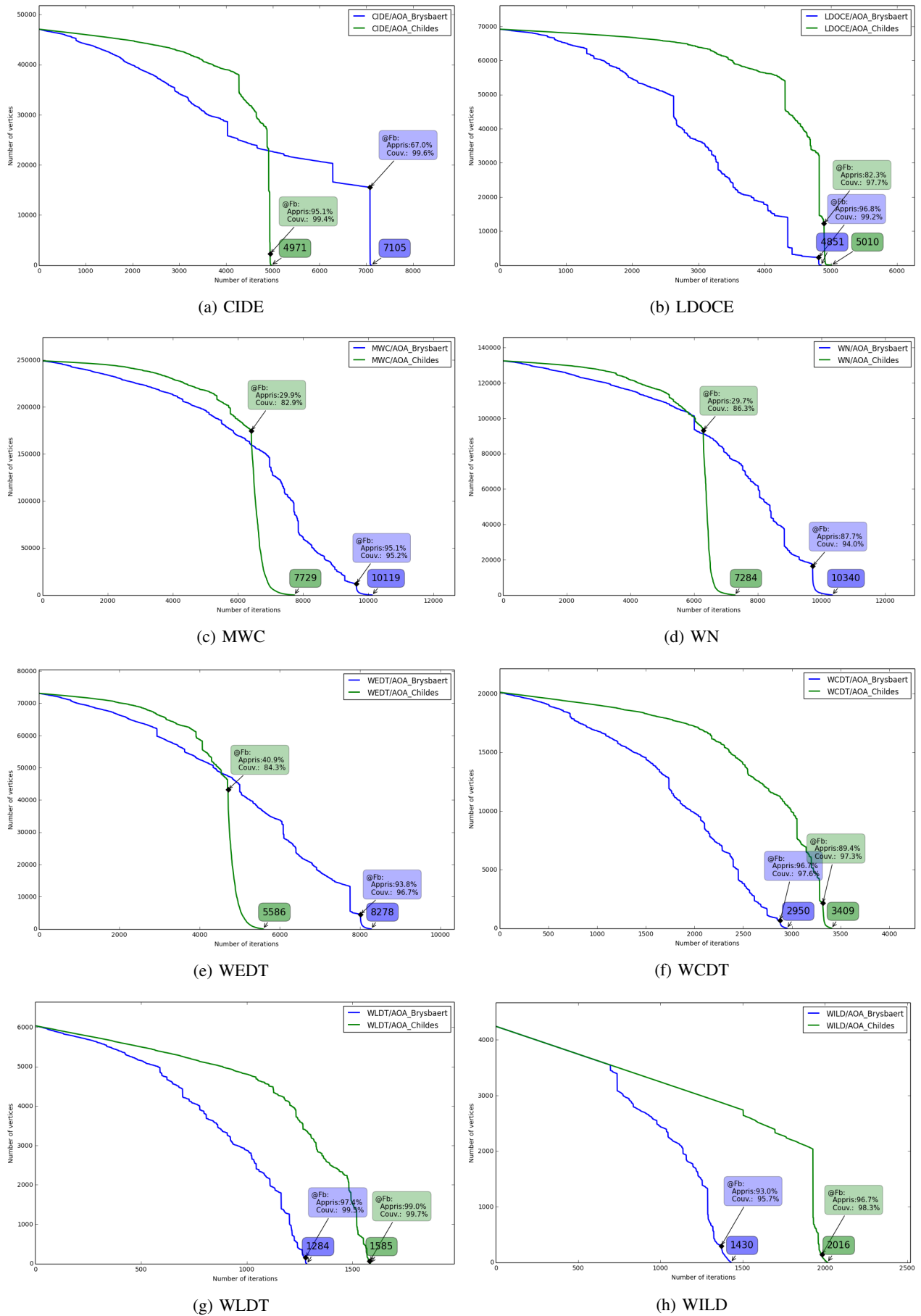


Figure 13. Learning: AOA based Strategies

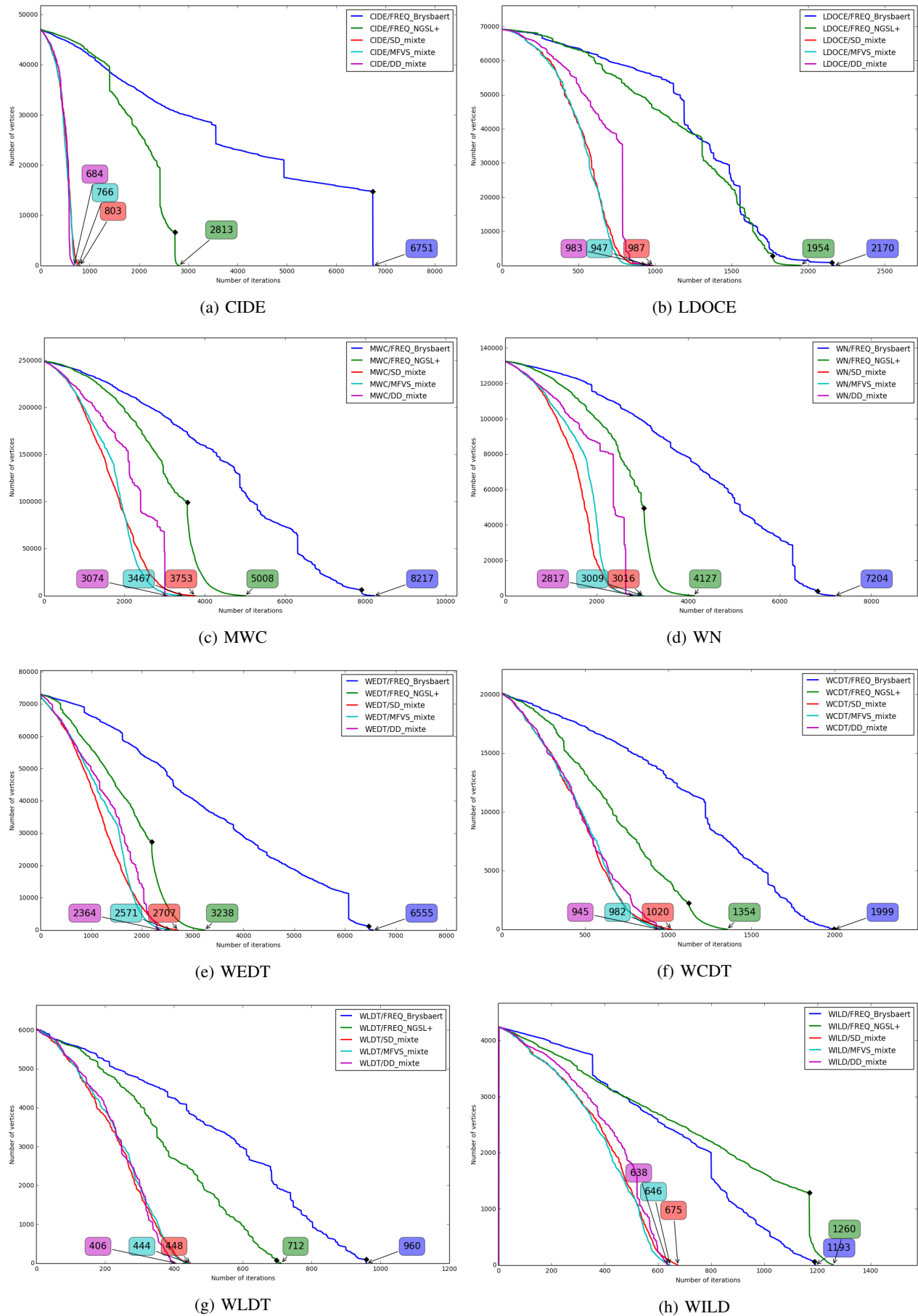


Figure 14. Learning: Optimized Strategies vs Mixed

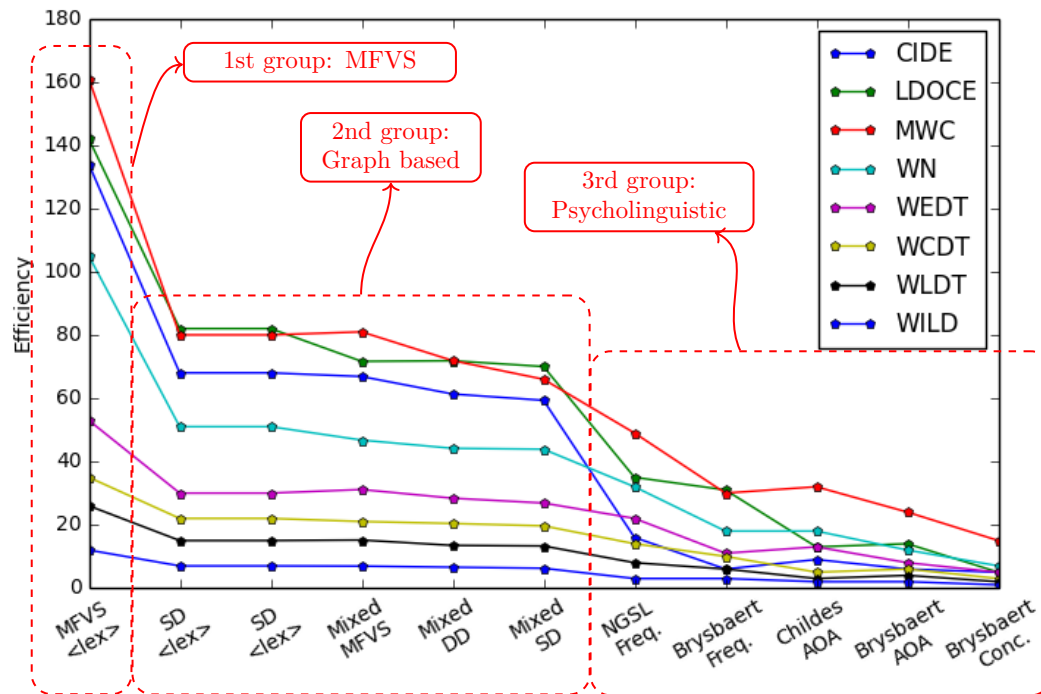


Figure 15. Lexicons: Efficiency vs Strategies

of their corresponding nodes, one can learn rapidly all the lexemes of a lexicon. In practical terms, this corresponds to a list of words ordered according to the number of times words appear in the definition of other words. Such a method by far outperforms psycholinguistic strategies .

In addition to their use as a dictionary analysis tool, the algorithmic strategies examined present additional advantages. To our knowledge, they represent a new approach for the development of word lists similar to those used in language teaching.

The word lists used by ESL teachers have traditionally been corpus-based, that is mainly built according to words frequency in a corpus. As an alternative, we propose a simple algorithmic strategy, based on the out-degree of vertices.

Although it is not possible to claim that the value of a word list is limited to its “efficiency”, we believe that this new approach could be used with profit, especially in cases where it is not possible to use existing word lists. In this case, or in the absence of an established corpus, the use of a digital lexicon or specialized dictionary would allow to establish easily a list of the relevant words or concepts, as well as the order in which they should be learned.

Future perspectives

A few words about the many research tracks left unexplored will conclude this article.

One might first think of extending the field of experimentation to new sources of data. Whether using new dictionaries, different digital lexicons, other algorithms for graph analysis, or new psycholinguistic norms, the possibilities are numerous. Similarly, one could explore dictionaries in fields such as medicine, mathematics, music or other specialized domains.

In addition, although resources in this area are often quite difficult to obtain, other languages would definitely offer rewarding research avenues. The analysis of monolingual dictionaries for languages other than English, or even of bilingual dictionaries, would for sure present many challenges.

Finally, the use of more advanced techniques to lexically disambiguate the definitions would offer a significant improvement to our methodology. Although the first sense heuristic usually gives satisfactory results and constitutes a strong baseline, newer techniques using neural networks and deep learning would certainly be worthwhile to explore. Improved word sense disambiguation as well as handling of compound lexical items would allow to build associated graphs more representative of underlying dictionaries and lexicons.

REFERENCES

- [1] J.-M. Poulin, A. B. Massé, and A. Fonseca, “Strategies for learning lexemes efficiently: A graph-based approach,” in *COGNITIVE 2018: The Tenth International Conference on Advanced Cognitive Technologies and Applications*. ThinkMind, 2018, pp. 18–23.
- [2] J.-C. Boulanger, *Les inventeurs de dictionnaires [ressource électronique] : De l'eduba des scribes mésopotamiens au scriptorium des moines médiévaux*. Canadian electronic library., ser. Collection Regards sur la traduction. Ottawa, Ont.]: Presses de l'Université d'Ottawa, 2003.
- [3] Merriam-Webster. (2018) Merriam-webster. [Online]. Available: <https://www.merriam-webster.com>
- [4] Collins. (2018) Collins dictionary. [Online]. Available: <https://www.merriam-webster.com>
- [5] G. Clark, “Recursion through dictionary definition space: Concrete versus abstract words,” *On WWW at http://www.ecs.soton.ac.uk/Åharnad/Temp/concreteabstract.pdf*. Accessed, vol. 23, no. 06, 2003.

- [6] P. Procter, *Longman Dictionary of Contemporary English (LDOCE)*. Essex, UK: Longman Group Ltd., 1978.
- [7] —, *Cambridge International Dictionary of English (CIDE)*. Cambridge University Press, 1995.
- [8] M. Steyvers and J. B. Tenenbaum, “The large-scale structure of semantic networks: Statistical analyses and a model of semantic growth,” *Cognitive science*, vol. 29, no. 1, pp. 41–78, 2005.
- [9] C. Fellbaum, Ed., *WordNet An Electronic Lexical Database*. Cambridge, MA ; London: The MIT Press, May 1998.
- [10] P. M. Roget, *Roget’s Thesaurus of English Words and Phrases...* TY Crowell Company, 1911.
- [11] A. Blondin Massé, G. Chicoisne, Y. Gargouri, S. Harnad, O. Picard, and O. Marcotte, “How is meaning grounded in dictionary definitions?” in *Proceedings of the 3rd Textgraphs Workshop on Graph-Based Algorithms for Natural Language Processing*. Association for Computational Linguistics, 2008, pp. 17–24.
- [12] O. Picard, A. Blondin-Massé, S. Harnad, O. Marcotte, G. Chicoisne, and Y. Gargouri, “Hierarchies in dictionary definition space,” *arXiv preprint arXiv:0911.5703*, 2009.
- [13] O. Picard, A. Blondin Massé, and S. Harnad, “Learning word meaning from dictionary definitions: Sensorimotor induction precedes verbal instruction,” 2010.
- [14] O. Picard, M. Lord, A. Blondin-Massé, O. Marcotte, M. Lopes, and S. Harnad, “Hidden structure and function in the lexicon,” *arXiv preprint arXiv:1308.2428*, 2013.
- [15] P. Vincent-Lamarre, A. B. Massé, M. Lopes, M. Lord, O. Marcotte, and S. Harnad, “The latent structure of dictionaries,” *Topics in cognitive science*, vol. 8, no. 3, pp. 625–659, 2016.
- [16] S. Harnad, “The symbol grounding problem,” *Physica D: Nonlinear Phenomena*, vol. 42, no. 1-3, pp. 335–346, 1990.
- [17] N. Schmitt, *Researching vocabulary: A vocabulary research manual*. Springer, 2010.
- [18] P. Prince, “Second language vocabulary learning: The role of context versus translations as a function of proficiency,” *The modern language journal*, vol. 80, no. 4, pp. 478–493, 1996.
- [19] N. Schmitt, “Instructed second language vocabulary learning,” *Language teaching research*, vol. 12, no. 3, pp. 329–363, 2008.
- [20] P. Joyce, “L2 vocabulary learning and testing: The use of L1 translation versus L2 definition,” *The Language Learning Journal*, pp. 1–12, 2015.
- [21] Merriam-Webster, *Merriam-Webster’s Collegiate Dictionary*, 11th ed., 2003.
- [22] J.-M. Poulin, “Stratégies efficaces pour l’apprentissage des mots d’un dictionnaire : Une approche basée sur les graphes,” Master’s thesis, Université du Québec à Montréal, 2018.
- [23] (2018). [Online]. Available: http://www.pearsonlongman.com/longman_france/pdf/dictionnaires.pdf
- [24] H. Jackson, *Lexicography: an introduction*. Routledge, 2013.
- [25] Merriam-Webster. (2018) Merriamwebsterthesaurus. [Online]. Available: <https://www.merriam-webster.com/thesaurus/>
- [26] D. A. Cruse, “The lexicon. the handbook of linguistics,” 2002, ed. by Mark Aronoff and Janie Rees Miller, ch10, Oxford: Blackwell.
- [27] A. Polguère, *Lexicologie et sémantique lexicale : notions fondamentales*, 3rd ed., ser. Paramètres. Les Presses de l’Université de Montréal, 2016.
- [28] N. Gader, S. Ollinger, and A. Polguère, “One lexicon, two structures: So what gives?” in *Seventh Global Wordnet Conference (GWC2014)*. Global WordNet Association, 2014, pp. 163–171.
- [29] Oxford. (2018) Oxford English dictionary. [Online]. Available: https://en.oxforddictionaries.com/definition/us/word_form
- [30] TLFi. (2018) Trésor de la langue française informatisé. [Online]. Available: <http://www.cnrtl.fr/definition/dictionnaire>
- [31] A. Spencer, “The handbook of linguistics,” in *The handbook of linguistics*, 1st ed., M. Aronoff and J. Rees-Miller, Eds. John Wiley & Sons, 2002, ch. Morphology, pp. 213–37.
- [32] C. Wenski-Béthoux, “Utilisation de produits multimédia pour la construction de compétences lexicale: analyse linguistique, psycholinguistique et didactique des apports des cédéroms, des sites internet et du travail en tandem pour l’apprentissage de l’allemand langue seconde,” Ph.D. dissertation, Lyon 2, 2005.
- [33] F. De Saussure, *Cours de linguistique générale: Édition critique*. Otto Harrassowitz Verlag, 1916 (1989), vol. 1.
- [34] O. Duchacek, “L’homonymie et la polysémie,” *Vox romanica*, vol. 21, p. 49, 1962.
- [35] D. Jurafsky and J. H. Martin, *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition*, 2nd ed., ser. Prentice Hall series in artificial intelligence. Pearson Prentice Hall, 2009.
- [36] G. A. Miller, “Dictionaries in the mind,” *Language and cognitive processes*, vol. 1, no. 3, pp. 171–185, 1986.
- [37] E. A. Corrêa, A. A. Lopes, and D. R. Amancio, “Word sense disambiguation: a complex network approach,” *Information Sciences*, 2018.
- [38] R. Navigli, “Word sense disambiguation: A survey,” *ACM Computing Surveys (CSUR)*, vol. 41, no. 2, p. 10, 2009.
- [39] R. V. Yampolskiy, “Turing test as a defining feature of ai-completeness,” in *Artificial intelligence, evolutionary computing and metaheuristics*. Springer, 2013, pp. 3–17.
- [40] P. van Sterkenburg, *A practical guide to lexicography*. John Benjamins Publishing, 2003, vol. 6.
- [41] Wordsmyth. (2017) Wordsmyth. [Online]. Available: <https://www.wordsmyth.net>
- [42] D. McCarthy, R. Koeling, J. Weeds, and J. Carroll, “Finding predominant word senses in untagged text,” *ACM*, 2004.
- [43] J. F. Sowa, *Knowledge representation logical, philosophical, and computational foundations*. Pacific Grove, Calif. ; Toronto: Brooks/Cole, 2000.
- [44] J. Hendler and F. van Harmelen, “The semantic web: webizing knowledge representation,” *Foundations of Artificial Intelligence*, vol. 3, pp. 821–839, 2008.
- [45] S. Russell, P. Norvig, and A. Intelligence, “Artificial intelligence, a modern approach,” *Artificial Intelligence. Prentice-Hall, Englewood Cliffs*, vol. 25, p. 27, 2010.
- [46] F. Lehmann, “Semantic networks,” *Computers & Mathematics with Applications*, vol. 23, no. 2-5, pp. 1–50, 1992.
- [47] D. L. Nelson, C. L. McEvoy, and T. A. Schreiber, “The university of south florida word association, rhyme, and word fragment norms,” 1999. [Online]. Available: <http://w3.usf.edu/FreeAssociation/>
- [48] J. A. Bondy, U. S. R. Murty *et al.*, *Graph theory with applications*. Oxford, UK: Elsevier Science Ltd., 1976.
- [49] V. V. Vazirani, *Algorithmes d’approximation. Traduction de: Algorithmes*, ser. Collection IRIS. Paris: Springer, 2006.
- [50] R. M. Karp, *Reducibility among Combinatorial Problems*. Boston, MA: Springer US, Miller, Raymond E. and Thatcher, James W. and Bohlinger, Jean D. editors, 1972, pp. 85–103.
- [51] H.-M. Lin and J.-Y. Jou, “On computing the minimum feedback vertex set of a directed graph by contraction operations,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, no. 3, pp. 295–307, 2000.
- [52] M. Lapointe, A. B. Massé, P. Galinier, M. Lord, and O. Marcotte, *Enumerating minimum feedback vertex sets in directed graphs*. LaBRI, Université Bordeaux 1, 2012, ch. Enumerating minimum feedback vertex sets in directed graphs, pp. 101–102.
- [53] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.
- [54] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of massive datasets*. Cambridge university press, 2014. [Online]. Available: <http://i.stanford.edu/~ullman/mmds.html>
- [55] S. Harnad, “Symbol-grounding problem,” *Encyclopedia of cognitive science*, 2003.
- [56] —, *To Cognize is to Categorize: Cognition is Categorization*. Elsevier, 2005.
- [57] A. L. Z. Monge, “L’évolution de l’enseignement du vocabulaire dans la classe de L2,” *Revista de Lenguas Modernas*, pp. 437–447, 2013.
- [58] I. S. Nation, *Making and using word lists for language learning and testing*. John Benjamins Publishing Company, 2016.

- [59] C. K. Ogden, "Basic English: A general introduction with rules and grammar, paul treber & co," *Ltd. London*, vol. 1940, 1930.
- [60] —. (2018) Ogden's basic english. [Online]. Available: <http://ogden.basic-english.org/wordmenu.html>
- [61] M. West, *A general service list of English words: with semantic frequencies and a supplementary word-list for the writing of popular science and technology*. Addison-Wesley Longman Limited, 1953. [Online]. Available: <http://jbauman.com/aboutgsl.html>
- [62] A. Coxhead, "A new academic word list," *TESOL quarterly*, vol. 34, no. 2, pp. 213–238, 2000.
- [63] V. Brezina and D. Gablasova, "Is there a core general vocabulary? introducing the new general service list," *Applied Linguistics*, vol. 36, no. 1, pp. 1–22, 2013.
- [64] C. Browne, "A new general service list: The better mousetrap we've been looking for," *Vocabulary Learning and Instruction*, vol. 3, no. 2, pp. 1–10, 2014.
- [65] C. Browne, B. Culligan, and J. Phillips. (2018) New general service list (ngsl). [Online]. Available: <http://www.newgeneralservicelist.org>
- [66] M. Brysbaert and B. New, "Moving beyond kučera and francis: A critical evaluation of current word frequency norms and the introduction of a new and improved word frequency measure for american English," *Behavior research methods*, vol. 41, no. 4, pp. 977–990, 2009.
- [67] K. Black, "Cambridge international dictionary of English," *The Booklist*, vol. 93, no. 19–20, June 1997.
- [68] Longman. (2018) Ldoce. [Online]. Available: <https://pearsonerpi.com/fr/elt/dictionaries/longman-dictionary-of-contemporary-english>
- [69] E. K. Brown and A. Anderson, Eds., *Encyclopedia of language and linguistics [ressource électronique]*, 2nd ed. Amsterdam: Elsevier, 2006. [Online]. Available: <https://www.sciencedirect-com.proxy.bibliotheques.uqam.ca:2443/science/referenceworks/9780080448541>
- [70] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, ser. NAACL '03. Stroudsburg, PA, USA: Association for Computational Linguistics, 2003, pp. 173–180.
- [71] V. Batagelj, A. Mrvar, and M. Zaveršnik, *Network analysis of dictionaries*. University of Ljubljana, Inst. of Mathematics, Physics and Mechanics, Department of Theoretical Computer Science, 2002.
- [72] E. W. Weisstein. (2003) Graph diameter. [Online]. Available: <http://mathworld.wolfram.com/GraphDiameter.html>
- [73] S. E. Schaeffer, "Graph clustering," *Computer science review*, vol. 1, no. 1, pp. 27–64, 2007.
- [74] A. Hagberg, P. Swart, and D. S. Chult, "Exploring network structure, dynamics, and function using networkx," Los Alamos National Lab.(LANL), Los Alamos, NM (United States), Tech. Rep., 2008.
- [75] P. Bonin, A. Méot, L. Aubert, N. Malardier, P. Niedenthal, and M.-C. Capelle-Toczek, "Normes de concrétude, de valeur d'imagerie, de fréquence subjective et de valence émotionnelle pour 866 mots," *L'année Psychologique*, vol. 103, no. 4, pp. 655–694, 2003.
- [76] K. J. Gilhooly and R. H. Logie, "Age-of-acquisition, imagery, concreteness, familiarity, and ambiguity measures for 1,944 words," *Behavior research methods & instrumentation*, vol. 12, no. 4, pp. 395–427, 1980.
- [77] M. Coltheart, "The mrc psycholinguistic database," *The Quarterly Journal of Experimental Psychology Section A*, vol. 33, no. 4, pp. 497–505, 1981.
- [78] M. Wilson, "Mrc psycholinguistic database: Machine-usable dictionary, version 2.00," *Behavior research methods, instruments, & computers*, vol. 20, no. 1, pp. 6–10, 1988.
- [79] T. A. Harley, *The psychology of language: From data to theory*. Psychology press, 2013.
- [80] A. Paivio, J. C. Yuille, and S. A. Madigan, "Concreteness, imagery, and meaningfulness values for 925 nouns," *Journal of experimental psychology*, vol. 76, no. 1p2, p. 1, 1968.
- [81] C. Browne, B. Culligan, and J. Phillips. (2018) New academic word list (nawl). [Online]. Available: <http://www.newacademicwordlist.org>
- [82] C. Browne and B. Culligan. (2018) Business service list (bsl). [Online]. Available: <http://www.newgeneralservicelist.org/bsl-business-service-list/>
- [83] —. (2018) Toeic service list (tsl). [Online]. Available: <http://www.newgeneralservicelist.org/toeic-list/>
- [84] V. Kuperman, H. Stadthagen-Gonzalez, and M. Brysbaert, "Age-of-acquisition ratings for 30,000 English words," *Behavior Research Methods*, vol. 44, no. 4, pp. 978–990, 2012.
- [85] B. MacWhinney, *The CHILDES Project: Tools for Analyzing Talk*. Mahwah, NJ: Lawrence Erlbaum Associates, third edition, 2000.
- [86] M. Brysbaert, A. B. Warriner, and V. Kuperman, "Concreteness ratings for 40 thousand generally known English word lemmas," *Behavior research methods*, vol. 46, no. 3, pp. 904–911, 2014.
- [87] R. A. Amsler, "The structure of the merriam-webster pocket dictionary," Ph.D. dissertation, The University of Texas at Austin, 1980.
- [88] D. Bullock, "Nsm+ ldoce: A non-circular dictionary of english," *International Journal of Lexicography*, vol. 24, no. 2, pp. 226–240, 2010.