Mixing Semantic Networks and Conceptual Vectors : Application to Hyperonymy

Violaine Prince, Mathieu Lafourcade

Mixing Semantic Networks and Conceptual Vectors
Application to Hyperonymy

Violaine Prince
LIRMM-CNRS and University Montpellier 2
161 rue Ada, 34392 Montpellier cedex 5
France
prince@lirmm.fr

Mathieu Lafourcade
LIRMM-CNRS and University Montpellier 2
161 rue Ada, 34392 Montpellier cedex 5
France
lafourcade@lirmm.fr

Abstract

In this paper, we focus on lexical semantics, a key issue in Natural Language Processing (NLP) that tends to converge with conceptual Knowledge Representation (KR) and ontologies. When ontological representation is needed, hyperonymy, the closest approximation to the is-a *relation, is at stake. In this paper we describe the principles of our vector model (CVM: Conceptual Vector Model), and show how to account for hyperonymy within the vector-based frame for semantics. We show how hyperonymy diverges from* is-a *and what measures are more accurate for hyperonymy representation. Our demonstration results in initiating a 'cooperation' process between semantic networks and conceptual vectors. Text automatic rewriting or enhancing, ontology mapping with natural language expressions, are examples of applications that can be derived from the functions we define in this paper.* **Keywords:** *knowledge representation, cognitive linguistics, natural language processing.*

1 Introduction

Natural Language Processing by machines (NLP) has long been a keystone for the branch of data processing that deals with Knowledge Representation (KR) and Artificial Intelligence (AI). Since language stands, for human beings, both as a formalism describing knowledge, and their favourite mean of communication, NLP has, for decades, acted as the test for intelligent processing. It is a NLP function that underlies the Turing test, i.e. the ability of mimicking humans in their means of communication. Thus, it is easy to show that NLP is one of the most fundamental topics in Cognitive Informatics.

Since the nineties, with the generalization of the world wide web, a new challenge bursted out, to be tackled by NLP researchers. A huge amount of textual data is now available to users, data they need to browse, understand, summarize and exchange. Therefore, to the problem of intelligence in communication, a new issue has been added to NLP topics: how to deal with important volumes of texts, that human users do not have the time or the power to analyze. New trends, arising from fields such as Information Retrieval (IR) and documents design, are now investigated by NLP techniques.

Within the wide NLP domain, *lexical semantics* are a key issue, since they represent the point of convergence with conceptual KR and ontologies extracted from web semantics. They also browse the area of lexical resources processing, so that many works in both NLP and AI have been devoted to lexical semantic functions, as a way to tackle the problem of word sense representation and discrimination. Among the well established trends in lexical semantics representations, two trends appeared to be conflictual, until now: the WordNet approach [13], [4], born from semantic networks, and KR-oriented, and the "vector approach", originated from the Saltonian representation in Information Retrieval (IR) [19], which has found a set of applications in NLP, especially with web semantics and documents design.

The first is based on logic and the second on vector-space algebra. The first is very efficient for *is-a* relationships (considered as the conceptual relation often embedded in hyperonymy) but is silent, or almost so, about several other interesting lexical functions such as antonymy [1] and thematic association[2]. Synonymy has been tackled by NLP researchers that enhanced the field of textual IR [21], [13], but discrimination between synonymy and hyperonymy has often led them to look for a more flexible notion such as semantic similarity [16].

The vector approach is completely at the opposite. Of-

---

[1] the opposition semantic relation. Example : 'big' and 'small' are related with antonymy. But so are 'moon' and 'sun' although they share many common traits.

[2] thematic association is often a 'loose' association of words or items belonging to the same topic, whatever the type of the relation.

fering very easily thematic association, it allows several distinct, fine-grained synonymy [8] and antonymy [22] functions to be defined and implemented, but is unable to differentiate or to valid the existence of hyperonymous relations.

In this paper, we show how to account for hyperonymy within the vector-based frame for semantics, relying on a cooperation between semantic networks and conceptual vectors, and how this can be applied to new functions such as word substitution, and semantic approximation, that belong to the field of semantic similarity. We use a semantic network to enhance vector learning, and symmetrically we build customized semantic networks out of hyperonymous relations between vectors. Experiments have been run on French, since our team owns a syntactic parser, and a semantic vectors producer for this language. For the time being, more than $200,000$ terms (words and expressions) are present in our lexical bases, and are regularly processed and tested with every tool we develop [3]. Of course, since methods are generic, they could be easily transposed to any language for which syntactic parsing and semantic vectors are provided[4]. Presenting and discussing our tool for hyperonymy is thus an important issue not only for this lexical base enhancement, but also for all applications that are derivable from semantic associations in texts.

2 Hyperonymy and *is-a* Relations

2.1 Defining Hyperonymy

Hyperonymy is a lexical function that, given a term t, associates to t one or many other terms that are more general, such as those used to define t in *genus* and *differentiae* (in the aristotelian definition). Its symmetrical function is called *hyponymy*. For instance, *bird* is a hyperonym for *sparrow*, *tit*,*eagle* and so forth. The latter are co-hyponyms of *bird*.

Hyperonymy, in almost all KR papers, is assimilated to the general argument of the *is-a* relationship (fundamentals are given in [1]). Let us remind that the *is-a* relationship is such as if X is a class of objects, and X' a subclass of X, then $is-a(X', X)$ is true. The rightmost argument X is called the *general* argument whereas X' is said to be the *specific* argument. The problem is that linguistic hyperonymy is not a "pure" *is-a* relation. When the word *horse* is defined, we find: "a herbivorous animal, with four legs, etc...". A good hyperonym for this definition of *horse* is *herbivorous mammal*. *Animal* is another hyperonym, since '*herbivorous mammal is-a mammal* and *mammal is-a ani-*

---

[3]our French lexical base and different tools provided for thematic association are all gathered at the following URL : http://www.lirmm.fr/∼ lafourca.

[4]for English, Roget-based vector representations are definitely adequate.

mal' is true. However, thematically, a *horse* is very close to a *herbivore*, whereas *herbivores* do not constitute a class but a set of individuals that may belong to different lines of the taxonomy (birds and insects and reptiles could be herbivorous, but also metaphorically, many other things). Thus, even if, in language, one wants to write that *a horse is a herbivore* eventhough *horse is-a herbivore* is false.

2.2 Some Specific Linguistic Issues Related with Hyperonymy

Linguistically, a *mammal* is not as good a hyperonym as *herbivorous mammal* for *horse*, because it is too vague. Too many mammals exist, and thus, the more precise the term, the better it is. *Mammal from the equine family* is precise but non informative to the plain user. If IR is stake, one would better be close to the language that is generally used. Thus, *herbivorous mammal* could appear as a trade-off. However, this can 'break' the *is-a* chain, because other relations can be mixed with the general argument. Here *herbivorous* acts as an attribute. But in itself, as a language item, *herbivores* exist as the set name of all animates that share this property. The status of the *attribution* relation is not well defined in all KR-derived models. In fact, attributes are termed as such as the result of the designer decision, and not because of their intrinsic properties.

In short, hyperonymy often appears as a complex function resulting from the composition of *is-a* and *is-attribute* relations, the latter originally present in the semantic networks model, but being abandoned by several formalisms, because of their ambiguous status.

The second linguistic problem is *polysemy*. A word is not a concept, it may address many concepts, and in many different ways with different intensities. A *horse* is:

- an animal

- a power unit for motors

- a mean of transportation.

The three 'points of view' over *horse* are not independent from each other. Historically, the animal has been ridden by humans and served as a mean of transportation. When shifting to mechanical devices, people needed to compare artificial modes of transportation and their original mean. Thus, they used the *horse* as a power unit as *candles* have been used as a mean of comparison for light intensity.

2.3 WordNet and Hyperonymy: How KR Tackles Linguistic Issues

WordNet is a built taxonomy of words, and as such, only captures *is-a* relations. Polysemous words having many definitions, and thus many hyperonyms, are tied with as

many *is-a* relations, which explains why WordNet is a network and not a tree. WordNet discards specific relations, and adresses polysemy only through the modelling of multiple inheritance in *is-a* chains: every step of the chain of classes and subclasses must verify the order relation. As language has not the same density of items everywhere, WordNet appears as a network with a certain amount of *gaps* in some locations and a fine-grained mesh in other places. For instance, the closer to the 'root', the more vague and scarce the words are. This property is important because, unlike local ontologies that are balanced in their densities, WordNet is closer to the core of problems that NLP has to deal with. Vagueness in IR, as well as in indexation, could be a very bad feature.

2.4 Hyperonymy and Word Definition

As shown before, hyperonyms could be extracted, when they are not known, from most dictionary-like definitions. Only general concepts, which tend to play the role of hyperonyms (and *is-a*) superclasses of many others, are not defined through aristotelian definition, but are explained by their hyponyms. This is why, in our CVM (Conceptual Vector Model) model presented in next section, we consider the existence of a "hyperonymy horizon" beyond which definitions become inversed: hyperonyms are more difficult to find and less explicative than hyponyms. The word *action* is almost at the top of the WordNet taxonomy and dictionary definitions tend to explain it with more specific words.

3 The Conceptual Vector Model (CVM)

Vectors have been used in Information Retrieval for long [20] and for meaning representation by the LSI model [3] from latent semantic analysis (LSA) studies in psycholinguistics. In NLP, and in the early nineties, [2] has provided a formalism for the projection of the linguistic notion of *semantic field* in a vector space, from which our model is inspired.

From a set of elementary notions, *concepts*, it is possible to build vectors (conceptual vectors) and to associate them to lexical items.[5] The hypothesis that considers a set of concepts as a generator to language has been long described in the Roget Thesaurus designed by Oxfordian Lexicologists at the end of the 19th century [18] (we call it the *thesaurus hypothesis*) and has been used by researchers in NLP (e.g. [23]) recently. Polysemous words combine different vectors corresponding to different meanings. This vector approach is based on well known mathematical properties: it is thus possible to undertake formal manipulations

attached to reasonable linguistic interpretations. Concepts are defined within a thesaurus (in our prototype applied to French, we have chosen [10] where 873 concepts are identified to compare with the 1043 provided by the Roget Thesaurus [18]). To be consistent with the thesaurus hypothesis, we consider that this set constitutes a generator 'family' for words and their meanings. This set is probably not free (no proper vectorial base)[6] and as such, any word would project its meaning on this space according to the following principle.

3.1 Principle

Let be $\mathcal{C}$ a finite set of n concepts, a conceptual vector V is a linear combination of elements c_i of $\mathcal{C}$. For a meaning A, a vector $V(A)$ is the description (in extension) of activations of all concepts of $\mathcal{C}$. For example, the different meanings of ‹*door*› could be projected on the following concepts (the set of pairs (CONCEPT[intensity]) are ordered by increasing values): V(‹*door*›) = (OPENING[0.3], BARRIER[0.31], LIMIT[0.32], PROXIMITY[0.33], EXTERIOR[0.35], INTERIOR[0.37], ...)

In practice, the largest $\mathcal{C}$ is, the finer the meaning descriptions are. In return, computer manipulation is less easy. As most vectors are dense (very few null coordinates), the enumeration of activated concepts is long and difficult to evaluate. We generally prefer to select the thematically closest terms, i.e., the *neighbourhood*. For instance, the closest terms ordered by increasing distance of ‹*door*› are: $\mathcal{V}$(‹*door*›)=‹*portal*›, ‹*portiere*›, ‹*opening*›, ‹*gate*›, ‹*barrier*›,...

To handle semantics within this vector frame, we use the common operations on vectors. An interesting measure is the angular distance that accounts for a *similarity measure*. As an example, we present, hereafter, the vector sum, the scalar product and the angular distance equations.

3.1.1 Vectors Sum

Let A and B be two vectors, we define V as their *normed sum*:

$$V = X \oplus Y \quad | \quad v_i = (x_i + y_i)/\|V\| \tag{1}$$

Intuitively, the vector sum of A and B corresponds to the union of semantic properties of A and B. This operator is idempotent as we have $A \oplus A = A$. The null vector $\vec{0}$ is a neutral element of the vector sum and, by definition, we have $\vec{0} \oplus \vec{0} = \vec{0}$.

---

[5]Lexical items are words or expressions which constitute lexical entries. For instance, ‹*car*› or ‹*white ant*› are lexical items. In the following we will sometimes use *word* or *term* to speak about a *lexical item*.

[6]Let us remind that a vectorial base is a set of generative and free vectors. Two vectors are said to be free if their vector product is equal to zero. A set of vectors is considered free, if each couple of vectors contained in it, is free.

3.1.2 Vectors Product

The vector product is equivalent to a *normed term to term product*. Let X and Y be two vectors, we define V as *their normed term to term product*:

$$V = X \otimes Y \quad | \quad v_i = \sqrt{x_i y_i} \qquad (2)$$

This operator is idempotent and $\vec{0}$ is absorbent.

$$V = X \otimes X = X \quad \text{and} \quad V = X \otimes \vec{0} = \vec{0} \qquad (3)$$

Also following an intuitive approach, the vector product of A and B represents the intersection of semantic properties of A and B. This is a crucial feature for hyperonymy since a hyperonym and its hyponym could be seen as one 'containing' the properties of the other. But it is also important in synonymy and may give hints about polysemous properties of some conceptual vectors (intersections with many different vectors). A better function for emphasizing intersection is given in the paragraph about contextualization.

3.1.3 Angular Distance

Let us define $Sim(A, B)$ as one of the *similarity* measures between two vectors A and B, often used in Information Retrieval. We can express this function as:

$$Sim(A, B) = \cos(\widehat{A, B}) = \frac{A \cdot B}{\|A\| \times \|B\|}$$

with "·" as the scalar product. We suppose here that vector components are positive or null. Then, we define an *angular distance* D_A between two vectors A and B as follows:

$$D_A(A, B) = \arccos(Sim(A, B))$$
$$\text{with} \quad Sim(A, B) = \cos(\widehat{A, B}) = \frac{A \cdot B}{\|A\| \times \|B\|} \qquad (4)$$

This function constitutes an evaluation of the *thematic proximity* as it measures the angle between the two vectors. We would generally consider that, for an angular distance $D_A(A, B) \leq \frac{\pi}{4}$, (i.e. less than 45 degrees), A and B are thematically close and share many concepts. For $D_A(A, B) \geq \frac{\pi}{4}$, the thematic proximity between A and B would be considered as loose. Around $\frac{\pi}{2}$, both vectors are othogonal, and thus tend to diverge very wildly. D_A is a real distance function. It verifies the properties of reflexivity, symmetry and triangular inequality. In the following, we will speak of *distance* only when these last properties will be verified, otherwise we will speak of *measure*.

3.1.4 Contextualization

When two terms are in presence of each other, some of the meanings of each of them are thus selected by the presence of the other, acting as a *context*. This phenomenon is called *contextualization*. It consists in emphasizing common features of every meaning. Let X and Y be two vectors, we define $\gamma(X, Y)$ as the contextualization of X by Y as:

$$\gamma(X, Y) = X \oplus (X \otimes Y) \qquad (5)$$

This function is not symmetrical, translating the non symmetry between the role of a context and the role of a contextualized term. As for other mathematical properties: the operator γ is idempotent ($\gamma(X, X) = X$) and the null vector is the neutral element. ($\gamma(X, \vec{0}) = X \oplus \vec{0} = X$). We will notice, without demonstration, that we have thus the following properties of *closeness* and of *farness*:

$$D_A(\gamma(X, Y), \gamma(Y, X))$$
$$\leq \{D_A(X, \gamma(Y, X)), D_A(\gamma(X, Y), Y)\} \qquad (6)$$
$$\leq D_A(X, Y)$$

The function $\gamma(X, Y)$ brings the vector X closer to Y proportionally to their intersection. The contextualization is a low-cost meaning of amplifying properties that are salient in a given context. For a polysemous word vector, if the context vector is relevant, one of the possible meanings is *activated* through contextualization. For example, *bank* by itself is ambiguous and its vector is pointing somewhere between those of *river bank* and *money institution*. If the vector of *bank* is contextualized by *river*, then concepts related to finance would considerably dim.

3.2 Implemented Lexical Functions: Synonymy and Antonymy

3.2.1 Synonymy

Two lexical items are in a synonymy relation if there is a semantic equivalence between them.

Synonymy is a pivot relation in NLP, but remains problematic, since semantic equivalence is not translatable into a mathematical equivalence relationship. It does not necessarily verify transitivity [12] and it could be, at least partially, confused with hyperonymy, when equivalence is reduced to semantic similarity [16]. A possible solution in a vector framework is to define a contextual synonymy (also proposed in [6]) represented by a three argument relation, which then supports the properties of an equivalence relationship. The suggested solution is called *relative synonymy* [8]. The functional representation is the following: a *relative synonymy* function Syn_R, is defined between vectors A, B and C, the later playing the role of a pivot, as:

$$Syn_R(A, B, C) = D_A(\gamma(A, C), \gamma(B, C))$$
$$= D_A(A \oplus (A \otimes C), B \oplus (B \otimes C)) \qquad (7)$$

The interpretation corresponds to testing the thematic closeness of two meanings (A and B), each one enhanced with what it has in common with a third (C). The advantage of such a solution is that it circumvents the effects of polysemy in cutting transitivity and symmetry. However, it does not provide a real distinction between a hyperonym of a given meaning of a word, and a true synonym of such a word. This problem is discussed in next section, when introducing more flexible notions such as *word substitution*.

3.2.2 Antonymy

Two lexical items are in antonymy relation if there is a symmetry between their semantic components relatively to an axis.

Three types of symmetry have been defined, inspired from linguistic research [14]. As an example, we expose only the 'complementary' antonymy proposed by [22]: The same method is used for the other types. *Complementary antonyms* are couples like *event/unevent, presence/absence*. Complementary antonymy presents two kinds of symmetry, (i) a value symmetry in a boolean system, as in the examples above, and (ii) a symmetry about the application of a property (*black* is the absence of color, so it is "opposed" to all other colors or color combinations). The functional representation is the following: The function $AntiLex_S$ returns the *n* closest antonyms of A in the context defined by C in reference to R. The partial function $AntiLex_R$ has been defined to take care of the fact that, in most cases, context is enough to determine a symmetry axis. $AntiLex_B$ is defined to yield a symmetry axis rather than a context. In practice, we have $AntiLex_B = AntiLex_R$. The last function is the *absolute antonymy function*. Their associated equations are given hereafter.

$$A, C, R, n \rightarrow AntiLex_S(A, C, R, n)$$
$$A, X, n \rightarrow AntiLex_R(A, X, n) = AntiLex_S(A, X, X, n)$$
$$\text{with} \quad X = (C|R) \tag{8}$$
$$A, n \rightarrow AntiLex_A(A, n) = AntiLex_S(A, A, A, n)$$

An implementation of these functions in the CVM is detailed and commented in [22]. Contrarily to synonymy, antonymy functions are modelled partially as semantic graphs and partially with conceptual vectors. Some oppositions are primarily of lexical nature, and can potentially be extended continuously in the meaning space.

3.3 Conceptual Vectors Construction

Building conceptual vectors is achieved through processing *definitions* from different sources (dictionaries, synonym lists, manual indexations, etc). Definitions are parsed with an NLP parser called SYGMART (available for French) and the corresponding conceptual vector is computed according to a procedure defined as follows.

After filtering according to various morphosyntactic attributes, we attach to the leaf (terminal node of the conceptual tree) a conceptual vector that is computed from the vectors of its k definitions. The most straighforward way (not the best) to do so is to compute the average vector: $V(w) = V(w.1) \oplus \cdots \oplus V(w.k)$. If the word is unknown (i.e. it is not in the dictionary), the null vector is taken instead.

Vectors are then propagated upward. Consider a tree node N with p dependants $N_i(1 \leq ip)$. The newly computed vector of N is the weighted sum of all vectors of N_i: $V(N) = \alpha_i N_1 \oplus \cdots \oplus \alpha_p N_p$. Weights α depend on the syntactic functions of the node. For instance, a *governor* [7] would be given a higher weight ($\alpha = 2$) than a regular node ($\alpha = 1$).The vectors computed for *a boat sail* and for *a sail boat* would not be identical. Once the vector of the tree root is computed a downward propagation is performed. A node vector is contextualized by its parent: $V'(N_i) = V(N_i) \oplus \gamma(N_i, N)$. This is done iteratively until reaching a leaf. This analysis method shapes, from existing conceptual vectors and definitions, new vectors. It requires a bootstrap with a kernel composed of pre-computed vectors, manually indexed for the most frequent or difficult terms and already defined in [10] . One way to build a coherent learning system is to take care of the semantic relations beetween items, and among them, synonymy, antonymy and the most important, hyperonymy. A relevant conceptual vector basis is obtained after some iterations in the learning process. At the moment of writing this article, our system counts more than $71,000$ items for French and more than $288,000$ vectors (because vectors may represent expressions and/or concepts). 2000 vectors are concerned with antonymy, and almost all of them are concerned with synonymy and hyperonymy. The computed functions have allowed to enhance the representation of almost all vectors.

3.4 Importance of Hyperonymy in CVM

A framework for hyperonymy is very useful for enhancing vector construction, since most vectors are built by parsing hyperonymous definitions provided by on-line sources on the Web. In fact, all lexical functions appear to be a great help for such as task. Symmetrically, relations between vectors are crucial for a data driven approach : trying to extract semantic relations in corpora ([23]) and thus building a domain ontology, or trying to organize information in corpora by relying upon *is-a* hierarchies ([11], [17]).

---

[7]the 'leader' in a syntactic group. For instance, subjects and verbs in a sentence are governors, whereas complements are definitely not. In a noun phrase, one of the nouns is a governor, and the other is a subordinate. Example : in the noun phrase 'grammar school' , 'school' is governor.

4 Computing Hyperonymy

As our approach is both data driven and hierarchy-based, we first try to define the impact of hyperonymy by measuring distances in corpora. These distances help to define *word substitution* and *semantic approximation* (with a taxonomical aspect). The theoretical model, both within semantic networks and vector space, is the *inclusion model*: a subclass includes the properties of its superclass. We show in this section how inclusion is dealt with and what results we have obtained.

4.1 Co-occurrence Model

Corpora are seen by researchers in NLP as set of real instantiations of linguistic phenomena, when compared to intentionally built toy sentences. The co-occurrence of items, either words or expressions, especially when it is repeated through a rich set of documents, is a good measure of a semantic relationship between these items [5]. This semantic relationship is sometimes assumed to be one of synonymy, closeness, but without a strict and rigorous linguistic definition. The Church's formula tends, however, to consider co-occurring items in a given string of words, and to rely on the frequency of this co-appearance to draw probabilities of relationship. What we suggest here, is to consider documents (and not pairs of items) as the unit measure, and a single co-occurrence in a document is as meaningful as repeated associations of the same items.

Thus, we define two measures of co-occurrence between a term w and an *hyperonym candidate* h:

$$M_T(w,h) = \frac{|H \cap W|}{|W|} \quad \text{and} \quad M_S(w,h) = \frac{|H \cap W|}{|H|} \tag{9}$$

W (resp. H) represents the set of documents in a given corpus that contains the term w (resp. h). $|W|$, respectively $|H|$, is the number of documents considered where w, respectively h appears. $|H \cap W|$ represents the set of documents that contains both terms h and w. M_T tends to determine the ratio of h and w co-occurrence as a pair, when compared to w. So if w is the reference element, and W is the relevant set of documents about w, then M_T tends to show how much of w meaning is available when using h, knowing that w and h do (or not) co-occur in texts. M_T is reminiscent of a *recall* measure in Information Retrieval. [8]

M_S on the contrary, relates the same numerator, with the number of documents containing h. So if h is a possible, but polysemous, hyperonym of w, or if h was scarcely related to w then $|H \cap W|$ would be small when compared to $|H|$, and M_S would define thus the relevance of replacing w by h, without bringing in irrelevant meanings or ideas. M_S is thus our realization of a *precision* measure. [9]

M_T and M_S are in an inverse relationship, but are neither symmetrical nor complementary. It is more a question of a trend.

4.1.1 Hyperonymy, Word Substitution, Taxonomy Evaluation

If we add the hypothesis that *h is possible hyperonym*, that is, we have good reasons to think that w *is-a h* is true, then the measure M_S evaluates to which extend w can be replaced by h and is thus a *word substitution measure*. Similarly, M_T is a taxonomy evaluation, the way one can approximate *horse* by *mammal* without being too vague.

We have run experiments by accessing Google *(www.google.com)* and the number of hits returned for each request. This number of hits corresponds to the cardinal of the considered set of documents. For example, we have the following result for the term *airplane*:

> aircraft /$M_T = 0.2659$ $M_S = 0.025$
> plane /$M_T = 0.1237$ $M_S = 0.1741$
> flying plane /$M_T = 0.5317$ $M_S = 0.0007$
> aircraft heavier than air /$M_T = 0.5238$ $M_S = 0.00004$

The best M_S value (when *airplane* is the reference) is for *plane*, however, it is small, probably because of the embedded polysemy in the term (it also means a flat world, a two dimensional mathematical space, …). In the general context of documents accessed by Google, people tend to use *plane* instead of *airplane*, when they exactly know what type of item they are talking about. However it has the worst value in the taxonomical evaluation: among the relevant hyperonym candidates, any other is more relevant than *plane*.

On the other side, *aircraft heavier than air* as well as *flying plane* have the best M_T or recall value. In fact, they are very good definitions or explanations of what is an *airplane*, eventhough people tend not to use them much as substitutes. This might appear strange, at least for *flying plane*: we interpret this absence of substitution frequency as the result of an economy principle that underlies most cognitive actions. If one undergoes the replacement of something by something else, one hopes at least to gain some cognitive effort. A shorter form as a substitution candidate is a good heuristic.

---

[8]Recall is the number of relevant items retrieved among the relevant records/documents present in the set of records/documents.

[9]Precision is the number of relevant items retrieved among the most exhaustive set of records/documents, where some are relevant and the others, not.

As a larger example, we have run the test for the term *horse*. We have found several meanings for *horses*:

- (a) the animal,

- (b) the class of horses or specie,

- (c) horse riding,

- (d) the representation of a horse,

- (e) the wooden horse,

- (f) the manlike women,

- (g) the power unit

- (h) an unreliable person

- …

The results of requests and co-occurrence measures are :

mammal / $M_T = 0.81$ $M_S = 0.0005$ (a)
animal / $M_T = 0.0986$ $M_S = 0.1523$ (a)
domestic animal / $M_T = 0.133$ $M_S = 0.0035$ (a)
kind of mammal / $M_T = 0.0481$ $M_S = 0.00002$ (a)
specie / $M_T = 0.1376$ $M_S = 0.0857$ (b)
horses / $M_T = 0.4673$ $M_S = 0.2954$ (b)
equitation / $M_T = 0.3498$ $M_S = 0.0991$ (c)
representation / $M_T = 0.0399$ $M_S = 0.0505$ (d)
toy / $M_T = 0.1363$ $M_S = 0.0184$ (e)
child toy / $M_T = 0.2387$ $M_S = 0.0004$ (e)
wooden horse / $M_T = 0.2025$ $M_S = 0.0012$ (e)
woman / $M_T = 0.0363$ $M_S = 0.4012$ (f)
manlike woman / $M_T = 0.5692$ $M_S = 0.00003$ (f) unit /
$M_T = 0.033$ $M_S = 0.0647$ (g)
arbitrary unit / $M_T = 0.067$ $M_S = 0.00004$ (g)
power unit / $M_T = 0.1042$ $M_S = 0.0003$ (g)

mammal is the most precise for the taxonomy (hyperonym used in definition) but *animal* is a better susbsitution term, eventhough it might not be a very good substitute (M_S around 15%). *specie* is too vague, when compared to *horses*. *child toy* has a best rendering of the meaning in item *(e)* than *toy* but is not as good as a substitute.

As we have noticed before, short terms are better substitutes, as representatives of the economy principle in linguistics. Taken out of their context, they might appear, from a taxonomic point of view, quite vague or ambiguous. However, since they are never isolated, their role as substitutes is not overburdened by polysemy.

Let us finally notice that if M_T values might sometimes come close to 0.8, this is never the case with M_S. Ratios for substitution continue to be very small. We have run the same experiments of many other words, and we have noticed the same difference in scale between the two measures.

4.1.2 Building and Upgrading a Local Possible *is-a* Hierarchy

A good M_T measure for a possible hyperonym helps to create a local *is-a* hierarchy by testing values from the most particular item up to the most general one. For instance, for *horse*, we can extract, directly from the text, the knowledge as a *horse is-a* a *mammal* is better than a *horse is-a* an *animal* on the taxonomical line. Since we can calculate and show that a *mammal is-a* an *animal* is true, then it is easy to create the following *is-a* line :
horse $\leq$ *mammal* $\leq$ *animal*
where $\leq$ represents an *is-a* relationship.
However, these different lines have to be merged, and moreover, sometimes, new meanings (unknown or not encountered before) have to be added to the existing structure, transforming it from a tree-like hierarchy into a plain graph. This graph plays the role of an *extracted semantic network*, at least one that has emerged from raw texts, vector forms and nothing else. Figure 1 shows a portion of the semantic network for *horse*.
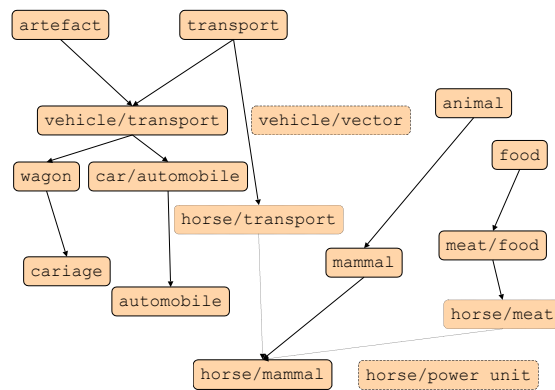

Figure 1. Hyperonym insertion in the built semantic network. Adding found hyperonyms can lead to the identification either of: (1) new salient properties in already existing meanings or (2)new meanings altogether. Thematic distance is used as a meaning selector.

About new meanings, in fact, two at least are lacking in the list of item given before.

- (i) a transportation mean (*we travelled on horseback*)

- (j) a type of food (horsemeat)

In this case, we do create the new meanings (*horse/ transportation mean* and *horse/meat*) and link them to their hyperonyms. The problem is that, starting from vectorized definitions, there is no way to catch these new meanings as they are not (yet) identified. Thus, to overcome this problem, we link each of these new meanings as hyperonym to

its closest already existing counterpart. In the above example, we have:

- *horse/ transportation mean* is closer to *horse/mammal* than to *horse/power unit*. This relation can be checked on their respective vector, and (sometimes) by pattern matching on some part of (encyclopedic) definition.

- *horse/meat* is closer to *horse/mammal* than to *horse/power unit*.

4.1.3 Conclusion about the Co-occurrence Model

These two measures, M_T and M_S, are particularly useful in semantic analysis. In fact, building a lexical network on the basis of M_T and M_S allows to recognize loose substitution hyperonyms (low M_T and high M_S). For example, during analysis, we can detect that the text thematic coherence is much stronger when we (re)subsitute *aircraft* to *plane*. Candidates for substitution are determined by the network structure strengthened by the angular distance between the candidate and the context. It is an iterated process that is globally converging ([9]). Thus, for textual analysis, we process in the reverse way of the text author, who has replaced precise terms with more or less vague hyperonyms, motivated by stylistic considerations (for example, deleting repetitions).

4.2 Inclusion Model

Inclusion, as a general idea, is what appears as common to both semantic networks in KR, and vector modelling in NLP when dealing hyperonymy. It is derived from a set theory approach, and suggests the following ;

If A is an hyperonym of B, then the properties of A are included in the properties B.
In KR, this means that A and B are in a super/subclass relationship (classical *is-a*). However, another definition also appears :
A is an hyperonym of B, if B has the same properties than A, and if B properties are instances of A properties
Examples:
'to cut' is a hyperonym of *'to saw'*. The latter provides the value of the action instrument (here the *saw*).
horses as the generic value of the specie, is a hyperonym of *horse* the individual (element (b) in the list of meanings for *horse*).

In KR this assets a set-member relationship (classical $member - of$), where the properties of A are instantiated by values belonging to the description of B.

As seen here, in fact, if KR tends to consider *is-a* hyperonyms only, unfortunately, NLP, at least in corpora, tends to consider also the $member - of$ relationship as a clue to a hyperonymy-hyponymy relationship. In fact, this is one of the cases where hyperonymy and hyponymy are symmetrical. In usage, if *to cut* acts as a good explanation of *to saw* the other way round is not true.

Thus, only in a restricted approach, the *is-a* and $member - of$ hyper/hyponymies are symmetrical. This symmetry, relevant to the Inclusion Model, dissapears in the Co-Occurrence Model (M_S is not equal to $1 - M_T$).

However, inclusion does exist, and could bring useful properties.

4.2.1 The Inclusion Measure

In a vector space approach, inclusion can be mesured through vector intersection and distance:

$$H(A, B) \Rightarrow$$
$$D_A(V(A), \gamma(V(A)\ V(B))) \quad (10)$$
$$\leq D_A(V(B), \gamma(V(A), V(B)))$$

For example, we have the following measure between *horse/mammal* and *mammal*:

$$D_A(V(\text{horse}), \gamma(V(\text{horse})\ V(\text{mammal}))) = 0.41$$
$$D_A(V(\text{mammal}), \gamma(V(\text{horse})\ V(\text{mammal}))) = 0.25$$

From this result, we deduce that *mammal* properties are included in *horse* Morover, if we know that *horse* and *mammal* are in a hyperonymic relation (either through a very good M_T value, or otherwise), then *mammal* is the hyperonym. The relationship between Inclusion and Co-occurrence Models is obvious : high M_T values for candidates provide an assumption about a good hyperonymic relationship, which in turn is cheched and thus validated (or invalidated) by the inclusion measure defined above.

4.2.2 Limits of the Inclusion Measure Scope

The model, restricted to the sole inclusion measure, operates very well for vectors that has been computed from hyperonymic definitions. But for very general terms, where definitions tends to be hyponymic (a collection of examples), the inclusion vector is reversed. More precisely, this is called the *horizon limit*. The *horizon* is constituted by leaves (terminal concepts) of the taxonomy on which the vector space is defined.

When the definition leads to a new vector, vectors of the terms present in this definition are mixed. Thus, the vector is flat compared with the main involved concept(s). We have a formal measure for *flatness* which is the *variation coefficient* V_C:

$$V_C(X) = \frac{s(X)}{\mu(X)}$$

$$\text{with} \quad s^2(X) = \frac{\sum(x_i - \mu(X))^2}{n} \quad (11)$$

V_C is the ratio between the standard deviation s of the vector component, and the mean μ. This a unitless value. By definition, V_C is only defined for non null vectors. If $V_C(A) = 0$ then the vector A is flat, that is, all components have the same value. At the maximum value of V_C (around 29 when $n = 873$), we have a boolean vector (only one component is activated with 1 while all others are zeros).
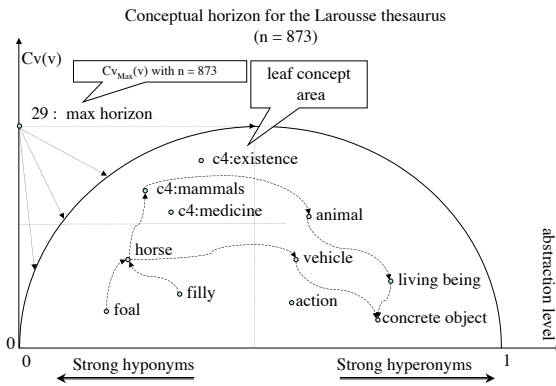

Figure 2. Graphical representation of the conceptual horizon. The horizon stands at the highest level of the variation coefficient which is the lowest level of the thesaurus hierarchy. On the left side, we have terms that are strictly specialization (by mixing) of concepts. On the right side, we have generalization of concepts, which similarly by vector mixing tend to lower the variation coefficient of vectors.

Over the horizon, we do have:

$$H(A, B) \Rightarrow$$
$$D_A(V(A), \gamma(V(A)\,V(B)))$$
$$\geq D_A(V(B), \gamma(V(A), V(B))) \quad (12)$$

4.2.3 The Conceptuality of a Vector : Beneath or Beyond the Concepts Hill ?

A very important issue is to be tackled: How is it possible to assess on which side of the *concepts hill* [10] a given vector stand? By itself, the variation coefficient just evaluates the general shape of the vector and its *conceptuality* relatively to the concept set. We have two ways to solve this problem:

---

[10] the graphical representation in the preceding figure shows a reversed parabol as a representation of the concept horizon, thus the metaphor of the 'hill' looks relevant.

1. the first is focusing on a lexical approach mixing lexical functions and information to vectors. The Co-occurence Model is a possible answer and, more generally, semantic graphs [11] as well. The Co-occurrence Model might be consolidated with an inclusion measure.

2. A second approach is to include, as a dimension of the vector space, every concept of the hierarchy and not only the leaves. This solution is only partial, because it cannot address the adjoining problem of polysemy when working on the lexical item level and not on the acception (conceptual) level.

We have undertaken the first approach, on a restricted scale (see discussion). The second one has been until now discarded, but before rejecting it completely, we would like to evaluate its true usefulness.
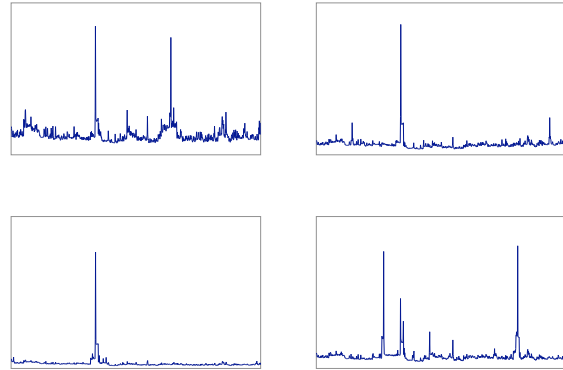

Figure 3. Graphical representations of the vectors of the terms ‹*poulain*› (in English *'foal'*), ‹*cheval*› (*'horse'*), ‹*Mammifères*› (*'mammals'*) and ‹*animal*›. The variation coefficient increases from left to right and top to bottom until the third vector (‹*Mammifères*›) (*'mammals'*) and then begins to decrease for the fourth one (‹*animal*›). Concepts are represented horizontally and their activation values vertically. If a not null vector is flat, then all concepts are equally activated. In this case the variation coefficient V_C is null.

4.3 Discussion

The experiments we have conducted (another example is given in the annex) on a collection of a few hundred nouns (and compound nouns), revealed the problem of the conceptual horizon. This horizon stands at the lowest level of the concepts hierarchy (in the hierarchy we use [10] for French language, which corresponds to the depth 4. For the Roget,

---

[11] among them, conceptual graphs or UNL based graphs are possible representations

this might go to depth 6 sometimes). Because of the nature of vector composition, the inclusion model should be inverted when terms stand beyond this horizon.

Detecting the conceptual horizon crossing is done through lexical models. More precisely, it can be achieved through the Co-occurence Model but also when identifying hyponyms. The detailed presentation of hyponyms identification is beyond the scope of this paper, but it is enough to say that more abstract terms (corresponding to large taxonomic classes) contain a large number of hyponyms. According to the Co-Occurence Model hyperonymy and hyponymy functions are not strictly symmetrical, both in their usage and behavior in corpora. In fact, if, in a semantic network in KR, hyperonymy and hyponymy are strictly symmetrical, language tends to assign different roles to hyperonyms and to hyponyms. For instance, if hyperonyms could be **good explanations through definitions**, hyponyms are **the best possible explanations through examples**. And very obviously, examples do not have the same relationship to assertion than definitions, and 'the best possible' is not even symmetrical to 'good'... However, both hyperonyms and hyponyms (of a given item) often co-appear in texts, and thus can be used together to strengthen the built network.

An application of our model, still under development, is a *paraphrase tool*, useful for stylistic goals. From a given text, the system produces a new text where terms are substituted by hyperonyms (or quasi synonyms). Initial results show that the most natural paraphrases are those which maximize the substitution value but not the taxonomic relevance. Such a tool could be used not only to globally assess the practical validity of our approach but also as a partial preprocess to Machine Translation.

5   Conclusion

In this paper we have tried to show how to account for hyperonymy within the vector-based frame for semantics, relying on a cooperation between semantic networks and conceptual vectors. After having assessed the importance of lexical functions such as synonymy and antonymy for lexical choice and conceptual vectors construction and usage, we have focused on hyperonymy, more difficult to discriminate in a numeric approach such as ours.

As our method is both data driven and hierarchy-based, we first tried to define the impact of hyperonymy by measuring distances in corpora. These distances help to define word substitution and semantic relevance (with a taxonomical aspect). The theoretical model, both within semantic networks and vector space, being the *inclusion model* we showed how inclusion has been dealt with and what results we have obtained.

Although being satisfactory, these results tend to reflect the multifaceted properties of hyperonymy: by being more complex than an *is-a* relation, hyperonymy needs to be constrained by the task to perform. If text correction or explanation are at stake, then *word substitution* is a good usage to apply hyperonymic properties. If taxonomy building is the goal, then *semantic relevance* is a better candidate. So, the same way other lexical functions such as synonymy and antonymy have been restricted by adding a notion of *relativity* when confronted to text bases, also hyperonymy appears not to be absolute, as the *is-a* relation is not either. It seems better to split it into its functions and to define it according to processing goals. Regarding applications, specific terminological database building as well as domain based ontologies for web browsing are achievable with semantic relevance. User-helping tools as linguistic assistance fit into the field of word substitution.

In a way, lexical functions, sometimes as theoretical as hyperonymy may appear to the non specialist, may have a great impact on NLP based tools for everyday assistance to computers users.

References

[1] Brachman R. J. and James G. Schmolze. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9(2):171–216, April–June 1985.

[2] Chauché J. Détermination sémantique en analyse structurelle : une expérience basée sur une définition de distance. *TA Information*, 31(1): 17–24.1990

[3] Deerwester S., S. Dumais, T. Landauer, G. Furnas, and R. Harshman, Indexing by latent semantic analysis. *Journal of the American Society of Information science*,416(6): 391–407,1990.

[4] Fellbaum C. (ed). WordNet: An Electronic Lexical Database. MIT Press, Cambridge, Massachussetts,1998.

[5] W. Gale and K. W. Church. *Identifying Word Correspondences in Parallel Texts.* Proceedings of the DARPA SNL Workshop. Asilomar, CA. 1991

[6] Gwei G. M. and E. Foxley. A Flexible Synonym Interface with Application examples in CAL and Help environments. *The Computer Journal* 30 (6): 551–557,1987.

[7] Hearst M. A. *Automated discovery of WordNet relations*, In C. Fellbaum ed. *WordNet : An Electronic Lexical Database* MIT Press, Cambridge, MA, 131–151, 1998.

[8] Lafourcade M. and V. Prince. *Relative Synonymy and Conceptual Vectors NLPRS01*, 127-134, 2001.

[9] Lafourcade M. *Conceptual Vectors and Fuzzy Templates for Discriminating Hyperonymy - is-a - and Meronymy - part-of - relations.* In proc. of OOIS 2003 Workshop MASPEGHI, P.Valtchev, M. Huchard, H. Astudillo (eds.), Montral, Canada October 6th 2003, ISBN 2-89522-035-2, pp. 19-29.

[10] Larousse, *Thésaurus Larousse - des idées aux mots - des mots aux idées.* Larousse, 1992.

[11] Lee J. H., M. H. Kim and Y. J. Lee. Information Retrieval based on conceptual distance in IS-A hierarchies. *Journal of Documentation*, 49(2), 188–207,1993.

[12] Lewis C. I. *The modes of meaning.* in Linsky ed, "Semantics and the philosophy of language". Urbana. NY, 1952.

[13] Miller G. A. and C. Fellbaum. Semantic Networks in English. in Beth Levin and Steven Pinker (eds.) *Lexical and Conceptual Semantics* , 197–229. Elsevier, Amsterdam, 1991.

[14] Palmer F. R. *Semantics: A New Introduction* . Cambridge University Press, 1976.

[13] Resnik P. *Using Information Contents to Evaluate Semantic Similarity in a Taxonomy*, *IJCAI-95*, 1995.

[16] Resnik P. Semantic similarity in a taxonomy:An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11, 95–130,1999.

[17] Resnik P. Disambiguating noun groupings with respect to WordNet senses. in S. armstrong, K. Church, P. Isabelle, E.Tzoukermann, S. Manzi and D. Yarouwsky (eds.) *Natural Language Processing using Large Corpora*, Kluwer Academic, Dordrecht, 1999.

[18] Roget P. M. *Thesaurus of English Words and Phrases* Longman, London, 1852.

[19] Salton G. *Automatic Information Organisation and Retrieval*, McGraw-Hill, New York, 1968.

[20] Salton G. and MacGill M.J.. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, 1983.

[21] Sparck Jones K. *Synonymy and Semantic Classification*. Edinburgh Information Technology Serie, 1986.

[22] Schwab D., M. Lafourcade and V. Prince. Antonymy and Conceptual Vectors. *COLING'02*, vol 2/2, 904-910 , 2002.

[23] Yarowsky D. Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora. *COLING'92*, 454–460, 1992.

6 Annex

Measuring M_T and M_S for the French term *peinture* :

art / $M_T = 0.133$ $M_S = 0.6913$ (a)
art de peindre / $M_T = 0.649$ $M_S = 0.0016$(a)
ouvrage / $M_T = 0.2248$ $M_S = 0.0955$ (b)
ouvrage d'un artiste / $M_T = 1.0$ $M_S = 0.00001$(b)
matière / $M_T = 0.2543$ $M_S = 0.1644$ (c)
produit / $M_T = 0.2301$ $M_S = 0.1755$ (c)
produit à base de pigments / $M_T = 1.0$ $M_S = 0.00004$ (c)
produit à base de pigments en suspension / $M_T = 1.0$ $M_S = 0.00004$ (c)
produit à base de pigments en suspension dans un liquide / $M_T = 1.0$ $M_S = 0.00004$ (c) couche / $M_T = 0.1443$ $M_S = 0.0876$ (d)
couche de couleur / $M_T = 0.4939$ $M_S = 0.0004$ (d)
description / $M_T = 0.2049$ $M_S = 0.1216$ (e)

The term *peinture* could be: (a) the *art*, (b) *painting*, (c) the *coloring matter*, (d) the *color layer*, and (e) a *description*. We can see that very precise terms are not good substitutes (see different cases for (c)). And inversely best substitutes are often more general and possibly polysemous terms.