To appear in
Computer Speech and Language
Special Issue on Word Sense Disambiguation

HyperLex:
Lexical Cartography for Information Retrieval[1]

Jean Véronis[2]
Equipe DELIC, Université de Provence, 29, Av. Robert Schuman, 13621 Aix-en-Provence Cedex 1, France

Abstract

This article describes an algorithm called *HyperLex* that is capable of automatically determining word uses in a textbase without recourse to a dictionary. The algorithm makes use of the specific properties of word cooccurrence graphs, which are shown as having "small world" properties. Unlike earlier dictionary-free methods based on word vectors, it can isolate highly infrequent uses (as rare as 1% of all occurrences) by detecting "hubs" and high-density components in the cooccurrence graphs. The algorithm is applied here to information retrieval on the Web, using a set of highly ambiguous test words. An evaluation of the algorithm showed that it only omitted a very small number of relevant uses. In addition, *HyperLex* offers automatic tagging of word uses in context with excellent precision (97%, compared to 73% for baseline tagging, with an 82% recall rate). Remarkably good precision (96%) was also achieved on a selection of the 25 most relevant pages for each use (including highly infrequent ones). Finally, *HyperLex* is combined with a graphic display technique that allows the user to navigate visually through the lexicon and explore the various domains detected for each word use.

Keywords

Word sense disambiguation, information retrieval, graphs, small worlds, graphic interface

1. Introduction

Keyword-based information retrieval on the Web, or any other large textbase, runs into the problem of the multiple uses of most words. The inescapable homography and polysemy of human languages generate considerable noise in the results. A query on the French word *barrage*, for example, may return pages on dams, play-offs, barriers, roadblocks, police cordons, barricades, etc. depending on the global frequencies and the particular ranking techniques used by search engines. Retrieving infrequent uses can prove quite tricky.

Of course, users can usually narrow down their queries by combining keywords with Boolean operators, but this is not always a straightforward task. Continuing with the above example, combining the word *barrage* with the word *match* does not necessarily produce all of the pages about "matchs de barrage" (play-offs): many pages about this topic do not contain the word *match*.

[1] This article is an extended version of a paper given at the *TALN'2003* conference (Véronis, 2003).

[2] *E-mail address* : Jean.Veronis@up.univ-mrs.fr

To get around this, one would have to list all lexical possibilities and write a query of the type *barrage* AND (*jouer* OR *jeu* OR *championnat* OR *rencontre* OR *football* OR *basket-ball* OR ...) (play OR game OR championship OR encounter OR soccer OR basketball OR ...), which is cumbersome and may still not produce the desired results. Besides, the general public is not very skillful when it comes to formulating such complicated queries. In a large-scale study on the *Excite* search engine, Spink, Wolfram, Jansen and Saracevic (2001) showed that less than 5% of the queries contained Boolean operators, and 50% were incorrect.[3] Less than 1% of the queries contained nested operators (as in the above example). Spink et al. even concluded:

> *"For an overwhelming number of Web users, the advanced search features do not exist. The low use of advanced search features raises questions of their usability, functionality, and even desirability, as currently presented in search engines."*

It thus seems worthwhile to carefully reconsider the applicability of word sense disambiguation methods to search engines. Within the past few years, an idea has been circulating that word sense disambiguation, and more generally natural language processing techniques or NLP, are useless in information retrieval (IR), and may even lower performance. I will show below that this claim rests on an erroneous interpretation of repeatedly cited articles like Voorhees (1999). The present study will hopefully demonstrate that this idea is false.

To be useful, a word sense disambiguation technique must exhibit sufficiently high performance. Many recent studies conducted in the Senseval competition (Kilgarriff, 1998) proposed substantial improvements in the available techniques and resources (see also Stevenson & Wilks, 2001). However, in my mind, one of the main problems in word sense disambiguation lies upstream, in the very sense lists used by systems. Conventional dictionaries are not suited to this task; they usually contain definitions that are too general (in our *barrage* example, the "act of blocking"), and there is no guarantee that they reflect the exact content of the particular textbase being queried. I showed experimentally that linguists have trouble matching the "senses" found in a dictionary and the occurrences found in a corpus (Véronis, 1998). What is more, textbase documents would still have to be automatically categorized on the basis of the dictionary "senses", an extremely difficult task that has eluded half a century of ongoing research efforts, and for which progress has been very recent (see Ide & Véronis, 1998, for a detailed description of the state of the art, and Stevenson & Wilks, 2001, for recent developments).

Schütze (1998) proposed a method based on "word vectors" that automatically extracts the list of "senses" (I prefer to speak of "uses") from a given corpus, while also offering a robust categorization technique. However, vector-based techniques come up against a major and highly crippling problem: large frequency differences between the uses of the same word cause most of the useful distinctions below the model's noise threshold to be thrown out.

In the present article, I propose a radically different algorithm, *HyperLex*, capable of automatically determining the uses of a word in a textbase without recourse to a dictionary. The algorithm exploits the specific properties of word cooccurrence graphs, which, as I will show below, turn out to be special graphs called "small worlds" (Watz & Strogatz, 1998; Barabási & Albert, 1999). Unlike the earlier word-vector methods, this approach can isolate highly infrequent uses (as rare as 1% of all occurrences) by detecting graph "hubs" and high-density components. The algorithm is applied here to information retrieval on the Web, using a set of highly ambiguous test words. An evaluation of the algorithm showed that it only omitted a very small number of relevant uses. In addition, *HyperLex* offers automatic tagging of word uses in context, with an excellent precision level (97% compared to 73% for baseline tagging, with a 82% recall rate). Remarkably good precision (96%) was also achieved when the 25 most relevant pages were selected for each use (including the highly infrequent ones). Finally, *HyperLex* comes with a graphic display technique that allows the user to visually navigate throughout the lexicon and explore the various domains detected for each use.

[3] My calculations based on Jansen, Spink, and Saracevic's tables (2000).

2. Past Research

Word sense disambiguation techniques were first applied to IR about thirty years ago, with Weiss's work (1973), but it was not until the 1990's that this type of application was tested in full scale (Krovetz & Croft, 1992; Voorhees, 1993; Wallis, 1993). The results obtained so far have been modest, and some studies have even reported a decline in performance. As mentioned above in the introduction, it was no doubt these studies -- especially the widely cited one by Voorhees (1999) -- that contributed to propagating the preconceived idea that word sense disambiguation, and NLP techniques in general, are useless or even detrimental in IR tasks. This idea is in fact a distortion of the results of studies published on the subject. If we read Voorhees (1999) more carefully, for example, we can see that she stresses the fact that performance is degraded under certain conditions by imperfect NLP techniques, and she is far from drawing the definitive conclusion that these techniques are of little interest to IR. Earlier, Sanderson (1994) had shown experimentally that with a correct disambiguation rate of 75% (typical of the state of the art at the time in matters of word sense disambiguation), performance in IR declined sharply, because the errors introduced by such a disambiguation system only make the initial ambiguity worse. Sanderson suggested that a 90% correct disambiguation rate would be necessary to improve IR performance. Not much later, Schütze and Pedersen (1995) used a disambiguator with 90% precision and did indeed find a 7 to 14% improvement in their querying system's performance.[4]

It is not trivial that the Schütze & Pedersen study (1995) -- one of the rare studies where word sense disambiguation was shown to have a positive effect in IR -- also happens to be one that did not make use of a dictionary containing a predefined list of senses: the "senses" were drawn directly from the corpus, using a method that will be described in detail below. In my mind, the use of a dictionary is the principal stumbling block of current disambiguation methods. I showed in a large-scale study conducted as part of the Senseval-1 evaluation exercise[5] (Véronis, 1998, 2001) that human annotators find it very difficult to perform the disambiguation process required of machines. Six linguistics students had to use the sense numbers supplied by a standard dictionary (*Petit Larousse*) to independently tag about 3700 occurrences of sixty polysemous words (20 adjectives, 20 nouns, and 20 verbs) in a corpus. A statistical analysis of the results showed that the pairwise interannotator agreement rate was mediocre: 41% for verbs and adjectives, and 46% for nouns (after factoring out the effect of chance). For certain words such as *correct, historique, utile, communication, degré, lancement,* and *station* (correct, historical, useful, communication, degree, launching, station), the tagging was virtually the same as it would have been if the subjects had responded at random. A detailed analysis of the problems encountered showed that in nearly every case, the dictionary entries did not contain enough surface cues to allow the annotators to reliably assign the occurrences in the corpus to the senses in the dictionary. To make things worse, the very division of dictionaries entries rarely takes into account the distributional constraints of the different word senses (number and types of complements, kinds of prepositions, selectional restrictions, etc.), and in many cases, it even contradicts these constraints. The lack of distributional cues and properties in most dictionaries leads to many vague definitions, particularly for abstract or highly polysemous words like *degré, économie, communication,* and *formation* (degree, economy/economics, communication, formation/training) which make up a large part of many texts (for a more detailed analysis, see Véronis, 2001).

This experimental finding is in line with what other researchers have continuously noted about dictionaries, although their studies have generally been more informal and less detailed (see for example Ahlswede, 1993, 1995; Ahlswede & Lorand, 1993; Amsler & White, 1979; Bruce & Wiebe, 1998; Jorgensen, 1990). This also applies to *WordNet*, despite its availability and widespread use in word sense disambiguation (see Fellbaum, Grabowski, & Landes, 1998): while offering a rich network of structured lexical information, the sense divisions it contains are in fact quite similar to those of a standard dictionary and therefore suffer from the same imperfections.

[4] An anonymous reviewer brought to my attention two recent studies which also hold a more positive view on the usefulness of WSD for IR (Gelbukh et al., 2003; Stokoe et al., 2003).

[5] The French part (see Segond, 2000).

As mentioned above, Schütze (1998) overcame this problem by automatically extracting the list of "senses" from the corpus itself. "Senses" correspond to clusters of similar contexts for a given word, and are thus defined in a distributional fashion. Although Schütze did not present things as such, his work takes up on an old idea. Traces of this idea are found back in Meillet (1926), for whom "the sense of a word can only be defined as an average of [its] linguistic uses." Wittgenstein (1953) argued for an analogous view in his *Philosophische Untersuchungen,* and Harris (1954: 155-158) adopted it in his linguistic program by defining meaning as a function of distribution. This idea also underlies Hornby's work (1942, 1954), which had a strong influence on British lexicography (this explains that British dictionaries are somewhat less subject to the criticisms I have outlined above). Whether the clusters thus identified actually constitute "senses" can probably be debated, so the more cautious term "use" will be employed here.

Schütze's implementation was derived from the vector-space model so well known in IR (see for example Salton & McGill, 1983). Each word is represented by a vector in a high-dimensionality Euclidian space, as are documents and queries in IR. The dimensions of the space are the different words that can occur in context with any word in the corpus, and the value of each component of the vector is the number of cooccurrences in a given context window. Illustrating again with the *barrage* example and outrageously reducing the possible contexts to two words only, *eau* (water) and *match* (match or game), for the sake of the example, the corresponding vector would be a representation in a two-dimensional space like the one shown in Figure 1.

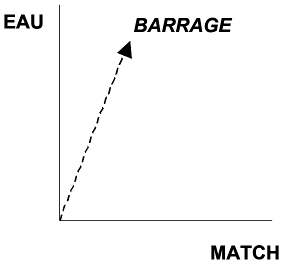

Figure 1. Vector of the word *barrage*

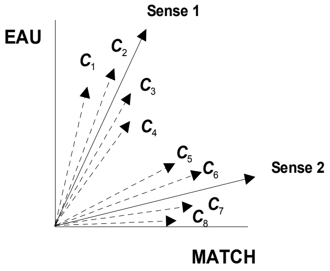

Figure 2. Context-vector

The vector representation of a word merges its different uses. Schütze defines context vectors in such a way that each one is the sum of the word vectors of words that occur in a given window for each particular context (Figure 2). A clustering algorithm detects coherent groups of contexts, each of which corresponds to a different sense of the concerned word. Obviously, such vector spaces have thousands of dimensions in reality, and Schütze uses a singular value decomposition technique that considerably reduces the dimensionality of the space (to only a hundred or so dimensions) before the clustering algorithm is applied.

The results of Schütze's experiment on a set of test words were very good, and as stated above, this technique significantly improves the performance of IR systems. However, in addition to its high cost in terms of computational resources, it has a major shortcoming: in my attempts to replicate the results, the clustering algorithm discriminated only uses that were few in number, more or less equiprobable and highly individualized. In fact, the words Schütze tested in his experiment met these criteria, that is, they were frequency-balanced homographs or near-homographs such as *plant*, *train*, *vessel*, and so on.

For most words, including the ones used in the present study like *barrage*, one or two main uses show up, followed by a variable number of low-frequency uses, more or less in accordance with a power law linking the rank of a use and its frequency (this was already apparent in Thorndike and Lorge's counts, 1938; see also Krovetz & Croft, 1992):

$$p(r) \propto r^{-\alpha} \qquad (1)$$

Figure 3 illustrates this for the various uses of the word *barrage*, observed in a corpus of five million words (based on data from Reymond, 2002).

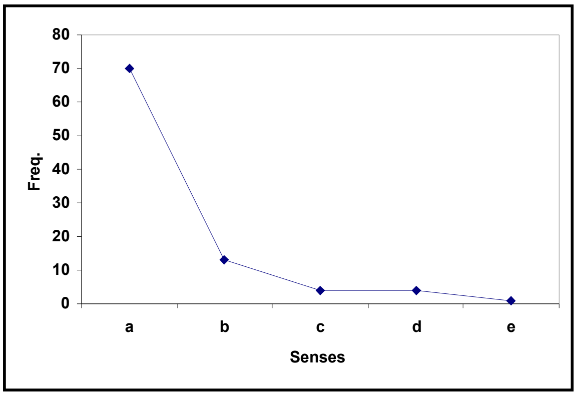

Figure 3. Corpus-based frequencies of the uses of *barrage*

But the low-frequency uses, like "match de barrage" (play-off) for example, are still not rare ones for an average speaker and are thus likely to be included in queries. Only at extremely low frequencies do the uses become uncommon -- e.g., *barrage* as guitar brace or opening bid in bridge, whose frequency cannot be precisely assessed. In fact, none of the uses of the test words employed in the present study can be considered "rare", even though many of them had a frequency of about 1%. My attempts to replicate Schütze's technique on these words totally failed, and this is what urged me to develop a method that would be less frequency-sensitive. It makes use of the particular properties of word cooccurrence graphs.

3. Small Lexical Worlds

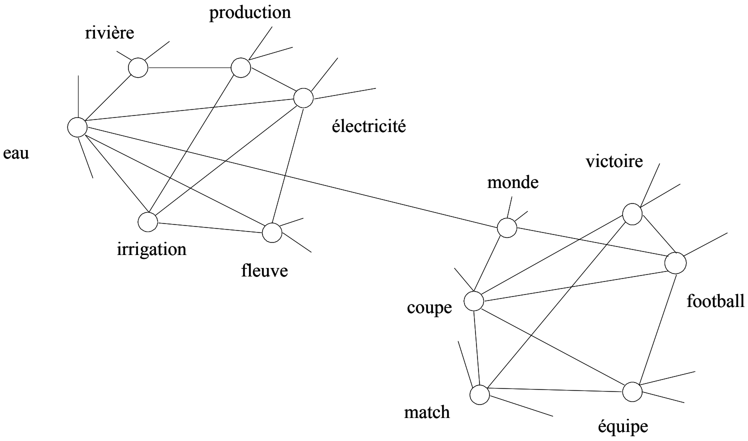

Figure 4. Graph of the cooccurrents of the French word *barrage*

One can construct a graph for each word to be disambiguated in a corpus (or *target word* -- details on how such a graph is generated will be presented later). The graph's nodes are the words that cooccur with the target word (in a window of a given size, e.g., a sentence or a paragraph). An edge connects two nodes, *A* and *B*, whenever the corresponding words are cooccurrent with each other. For instance, in the graph of the target word *barrage* (Figure 4), the nodes corresponding to *production* and *électricité* will be connected to each other because they occur together in contexts such as:

> *Outre la* ***production*** *d'****électricité****, le* ***BARRAGE*** *permettra de réguler le cours du fleuve...* (In addition to the **production** of **electricity**, the **DAM** will help regulate the river flow ...)

We shall see below that such graphs have the properties of the "small worlds" described by Watts and Strogatz (1998), one of today's key areas of research in graph theory. While a large part of the graph-theory studies have dealt with regular graphs or random graphs, the Watts and Strogatz study (1998), along with an ever-growing body of research, has shown that most real-world graphs and networks do not fall into either of these categories, but are in an intermediate state somewhere between order and chaos.

3.1. Properties of Small World Graphs

Watts and Strogatz (1998) defined two measures for characterizing small worlds: the *characteristic path length* (L) and the *clustering coefficient* (C). L is the mean length of the shortest path between two nodes of the graph. Let $d_{\min}(i, j)$ be the length of the shortest path between two nodes, i and j, and let N be the total number of nodes:

$$L = \frac{1}{N} \sum_{i=1}^{N} d_{\min}(i, j) \tag{2}$$

For each node i, one can define a local clustering coefficient C_i equal to the proportion of connections $E(\Gamma(i))$ between the neighbors $\Gamma(i)$ of that node. For a node i with 4 neighbors, for instance, the maximum number of connections is $\binom{|\Gamma(i)|}{2}$ = 6. If 5 of these connections actually exist, C_i = 5/6 ~ 0.83. The global coefficient C is the mean of the local coefficients:

$$C = \frac{1}{N} \sum_{i=1}^{N} \frac{|E(\Gamma(i))|}{\binom{|\Gamma(i)|}{2}} \tag{3}$$

This coefficient ranges between 0 for a totally disconnected graph and 1 for a complete graph.

In the case of a random graph of N nodes whose mean degree is k (mean number of edges per node or E/N, where E is the number of edges in the graph):

$$L_{\text{rand}} \sim \log(N) / \log(k) \tag{4}$$

$$C_{\text{rand}} \sim 2\, k / N \tag{5}$$

For example, a random graph of 1000 nodes and 10000 edges will have a mean degree k of 10, a characteristic path length L_{rand} of about log(1000)/log(10) = 3 and a clustering coefficient C_{rand} of about 10/1000 = 0.01.

For Watts and Strogatz (1998), small world graphs are characterized by the relations:

$$L \sim L_{\text{rand}} \tag{6}$$

$$C >> C_{\text{rand}} \tag{7}$$

Relation (5) means that at a constant mean degree, the number of nodes can increase exponentially, whereas the characteristic path length will only increase in a linear way. This accounts for the phenomenon observed by Milgram (1967), who first proposed the term "small world": any individual on the planet is only "six degrees away" from any other individual in the graph of social relations, even though there are several billion inhabitants.

Equation (6) indicates the difference between a small world and a random graph: in a small world, there will tend to be "bundles" or highly interconnected groups. Taking the social-relations example again, the friends of a given individual will be much more likely to be acquainted with each other than would be predicted if the edges of the graph were simply drawn at random.

Following the Watts & Strogatz article (1998), small worlds became a widely studied topic, and this type of structure was discovered in a large number of real networks including the Web, the Internet, networks of mathematicians who had cosigned an article, or actors in the same film, electrical distribution networks, protein interaction networks, and so on (see Newman, 2003). The distribution of node degrees has also been thoroughly examined: while in a random graph, the probability $p(k)$ that a node will be of degree k decreases exponentially with k (Poisson's law), most observed small worlds abide by a power law (Barabási & Albert, 1999):

$$p(k) \propto k^{-\alpha} \tag{8}$$

with α close to one.

Graphs that obey this law are called scale-free.[6] In a graph of this type, most nodes turn out to have few connections, while a very small number of nodes (hubs) are highly connected to a very large number of others. Deleting hubs can cause considerable damage to the network. The Internet is a typical example of such a graph.

3.2. Building Cooccurrence Graphs

Ten highly polysemous words were selected from among the ones that had caused substantial problems for the human annotators in Véronis (1998) (Table 1). For each word, a subcorpus of Web pages was compiled using the meta-engine Copernic Agent[7], and querying first with the singular form of the word and then with the plural form. Among the pages obtained, ones that occurred twice and ones that did not contain the word in question (errors of the type "Page not found", for example) were eliminated.

The paragraphs containing each target word were extracted and tagged using Cordial Analyseur[8], supplemented with some post-processing programs. Only nouns and adjectives were kept. At first, verbs were also retained, but they ended up causing a notable decline in performance since too many verbs like *commencer* (start), *pouvoir* (can) have very general uses. This is a temporary solution, of course, and this problem will be attacked in future research. The paragraphs were then filtered to eliminate function words (determiners, prepositions, etc.) as well as a certain number of general words found on a stoplist, especially words related to the Web itself, given this particular application (*menu, home, link, http,* etc.)[9] Words with less than 10 occurrences in the entire subcorpus were also discarded. Finally, contexts containing fewer than 4 words after filtering were deleted.

The cooccurrence matrix was generated from this filtered set of contexts: two words appearing in the same paragraph were said to cooccur.[10] Only those cooccurrences with a frequency of 5 or more were retained.

Table 1 gives the quantitative characteristics of the subcorpus collected for each word, and of the cooccurrence graph generated from it.

[6] By contrast, random graphs have a scale, their mean degree k, which is the peak of the degree distribution.

[7] http://www.copernic.com

[8] Developed by Synapse Développement : http://www.synapse-fr.com

[9] It is extremely important that the lemmatization and filtering processes be of high quality. If the method is robust as a whole, systematic lemmatization errors on the hubs of a graph can lead to disastrous results, as can the presence of unfiltered words like *menu* or *home*, which create artificial hubs that are unrelated to the domains of the subcorpus in question. The precision of the morphosyntactic tagging obtained here for the major grammatical categories (noun, adjective, verb) was about 99% (the fact that the main difficulties concerned distinctions between minor categories like prepositions and adverbs was helpful).

[10] Other possibilities could be explored, such as the utilization of a fixed-size window throughout the text. However, the paragraph seems to be a good contextual unit for use in real-world applications because a single cooccurrence matrix can be constructed for the entire corpus. The same matrix would be valid for all the words to be disambiguated, which would save a lot of processing time.

Target word	Translation	No. of Pages		No. of Contexts	
		Raw count	Useful	Raw count	Useful
BARRAGE	*dam, blockade, barrage...*	1702	1372	7256	6924
DETENTION	*detention, possession, holding, custody…*	2112	1270	8902	8728
FORMATION	*training, formation*	5974	1590	5248	4885
LANCEMENT	*launching, starting up, throwing...*	2828	1231	3307	3174
ORGANE	*organ, instrument, medium, representative…*	2786	994	2953	2849
PASSAGE	*passage, way, crossing, transition, coming by…*	3512	1046	4210	3894
RESTAURATION	*restoration, rehabilitation, catering, food industry…*	5327	1227	3522	3287
SOLUTION	*solution, answer*	6287	896	2085	1915
STATION	*station, halt, site…*	7916	1093	3837	3671
VOL	*flight, gliding, theft, robbery…*	5237	818	3001	2579

Table 1. Target words and quantitative characteristics of the subcorpora

3.3. Weighting

Each edge is assigned a weight that decreases as the association frequency of the words increases:

$$w_{A,B} = 1 - \max[\, p\,(A \mid B), p(B \mid A)\,] \qquad (9)$$

where $p(A \mid B)$ is the conditional probability of observing A in a given context, knowing that that context contains B, and inversely, $p(B \mid A)$ is the probability of observing B in a given context, knowing that it contains A. These probabilities are estimated from frequencies:

$$p(A \mid B) = f_{A.B} \,/\, f_B \quad \text{et} \quad p(B \mid A) = f_{A.B} \,/\, f_A \qquad (10)$$

In illustration, take the cooccurrences *eau* - *ouvrage* (water - work) and *eau* - *potable* (water - drinkable). Table 2 gives the number of contexts in which these word pairs appear together or separately in the *barrage* subcorpus. We can see that all occurrences of the word *potable* appear in conjunction with the word *eau*, whereas this is true of only some of the occurrences of the word *ouvrage*.

	EAU	~EAU	Total
OUVRAGE	183	296	479
~OUVRAGE	874	5556	6430
Total	1057	5852	6909

	EAU	~EAU	Total
POTABLE	63	0	63
~POTABLE	994	5852	6846
Total	1057	5852	6909

Table 2. Number of cooccurrences of *eau-ouvrage* (water - work) and *eau* - *potable* (water - drinkable)

This gives us:

$p(eau \mid ouvrage) = 183/479 = 0.38$ $p(ouvrage \mid eau) = 183/1057 = 0.17$ $w = 1 - 0.38 = 0.62$

$p(eau \mid potable) = 63/63 = 1$ $p(potable \mid eau) = 63/1057 = 0.06$ $w = 1 - 1 = 0$

This measure thus reflects the magnitude of the semantic "distance"[11] between words: when it is equal to 0, the words are always associated (with its highest possible value being equal to the frequency of the less frequent of the two words); when it is equal to 1, the words are never associated.

Edges with a weight above 0.9 are arbitrarily eliminated. This thresholding process is critical because it allows only those edges representing strong associations to be included in the graph.

[11] *w* is not a distance in the mathematical sense of the term, but a dissimilarity, since the triangular inequality does not hold.

Without it, the graph would tend to become totally connected as the corpus grows in size, due to the increasingly likely presence of accidental cooccurrences of any two word pairs.

Once the edges are weighted, a weighted clustering coefficient C' can be defined:

$$C'=\frac{1}{N}\sum_{i=1}^{N}\frac{\sum_{j=1}^{|E(\Gamma(i))|}(1-w_{ij})}{\binom{|\Gamma(i)|}{2}} \tag{11}$$

This coefficient is a little finer than the one originally proposed by Watts and Strogatz (1998): instead of merely reflecting the presence or absence of an edge, it also takes their respective weights into account.

3.4. Properties of Cooccurrence Graphs

After these various operations, the graphs obtained had the features listed in Table 3.

Word	***N***	***E***	***k***	***C***	***L***	C_{rand}	L_{rand}
BARRAGE	1203	6138	5.1	0.47	3.5	0.008	4.4
DETENTION	1418	19007	13.4	0.55	3.3	0.019	2.8
FORMATION	542	1531	2.8	0.44	3.5	0.010	6.1
LANCEMENT	617	2521	4.1	0.52	3.6	0.013	4.6
ORGANE	531	1997	3.8	0.44	4.0	0.014	4.7
PASSAGE	797	2916	3.7	0.47	4.5	0.009	5.2
RESTAURATION	512	1398	2.7	0.46	4.0	0.011	6.2
SOLUTION	253	1704	6.7	0.57	2.1	0.053	2.9
STATION	487	971	2.0	0.43	3.7	0.008	9.0
VOL	259	719	2.8	0.48	2.7	0.021	5.4

Table 3. Graph features

One can see that relations (5) and (6) are both obeyed, which means that the cooccurrence graphs are small world graphs. Also, the relation between $p(k)$ and k is approximately governed by a power law, as Figure 5 shows for the word *barrage*. The cooccurrence networks are scale-free, so they contain a small number of highly connected hubs and a large number of weakly connected nodes.

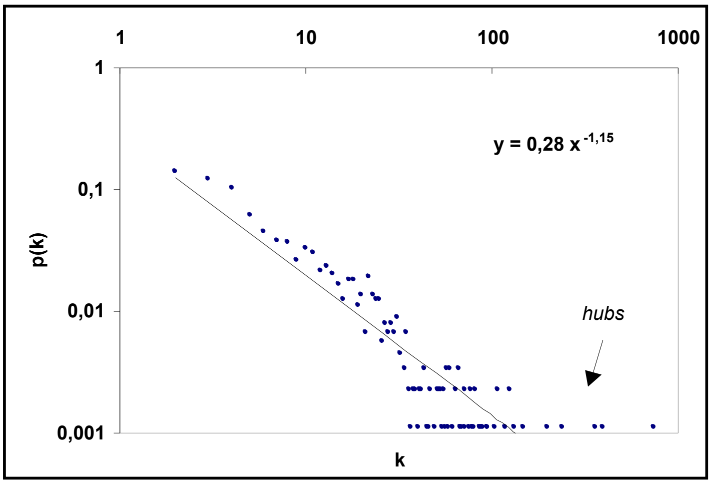

Figure 5. Power law on degrees, for the French word *barrage*

Finally, we find that a word's degree and frequency are highly correlated, in a nearly linear fashion (Figure 6). This property will be put to use in simplifying certain calculations.

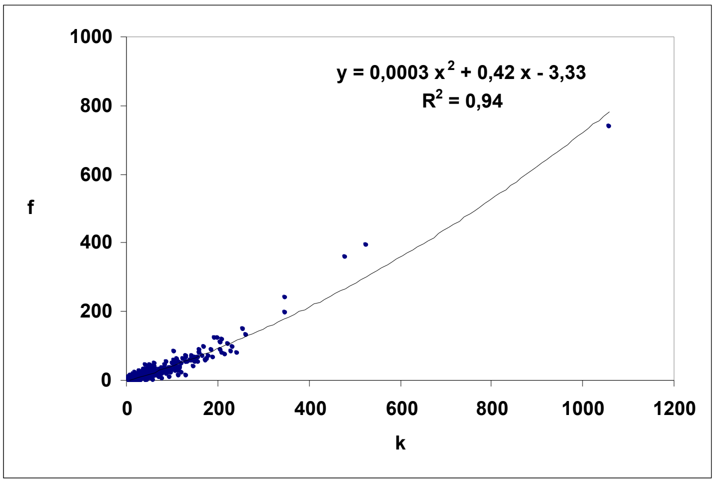

Figure 6. Correlation between degree and frequency (for *barrage*)

4. Detection of High-Density Components

The basic assumption underlying the method proposed here is that the different uses of a target word form highly interconnected "bundles" in a small world of cooccurrences, or in terms of graph theory, *high density components.* Accordingly, *barrage* (in the sense of a hydraulic dam) must cooccur frequently with *eau, ouvrage, rivière, crue, irrigation, production, électricité* (water, work, river, flood, irrigation, production, electricity), etc., and these words themselves are likely to be interconnected (Figure 4). Similarly, in the play-off use, *barrage* must cooccur frequently with *match, équipe, coupe, monde, football, victoire* (match, team, cup, world, soccer, victory), etc., which again are highly interconnected. Given the complexity of the language (particularly the fact that the cooccurrents are themselves ambiguous), there are also connections between components, which prohibits the use of algorithms that detect highly connected components or cliques, but the component interconnections must be few in number and have heavy weights.

Detecting the different uses of a word thus amounts to isolating the high-density components in its cooccurrence graph. Unfortunately, most exact graph-partitioning techniques are NP-hard, so, given that the graphs obtained have several thousand nodes and edges, only approximate heuristic-based methods can be employed. The detection of high-density components is a very popular area of research now being applied to the detection of "communities" or "authorized sources" on the Web, and to parallel computing. A drawback of the techniques developed in these areas is that they are not directly exploitable, because the heuristics are application-specific and depend on the particular properties of the graphs being analyzed.

The present algorithm makes use of the properties of small worlds and the scale-free feature demonstrated above. It consists of two steps. First, a certain number of hubs that will act as "roots" for the different components are detected. Then, the nodes that belong to each of these components are listed. The first step suffices to list the target word's uses in the corpus; the second is required for disambiguation and display.

4.1. Detecting Root Hubs

The starting point here is the observation that in every high-density component, one of the nodes has a higher degree than the others; it will be called the component's root hub. For example, for the most frequent use of *barrage* (hydraulic dam), the root hub is the word *eau* (water). It is easy to find, since it is the hub with the highest degree in the graph (and it is also the most frequent word).

Then the root hub of the next component is identified. The graph's structure, which consists of "bundles" with many internal connections but few connections to each other, enables us to apply a simple strategy: if the root hub just isolated is deleted, *along with all of its neighbors,* the chances of almost entirely eliminating the first high-density component are great. The very organization of small worlds is such that if a word of a reasonably high degree is part of the first component, it also has multiple connections to the nodes of which it is composed, and it is also very likely to be connected to the root hub. If not, one can be reasonably sure that it is part of some other component.

This strategy obviously deletes nodes that are not part of the first component. In the Figure 4 example, the *monde* node (world) will be deleted even though it is part of the *match de barrage* component. The assumption here is that because these intercomponent links are scarce, there will still be a sufficient number of nodes *specific to the second component* (in particular, its root hub) to ensure its detection.

The algorithm continues to iterate this process. The next hub candidate is *routier* (road-related), itself linked to *véhicule* (vehicle), *camion* (truck), etc. It is deleted, along with its neighbors, and so on, until no graph nodes are left (Figure 7).

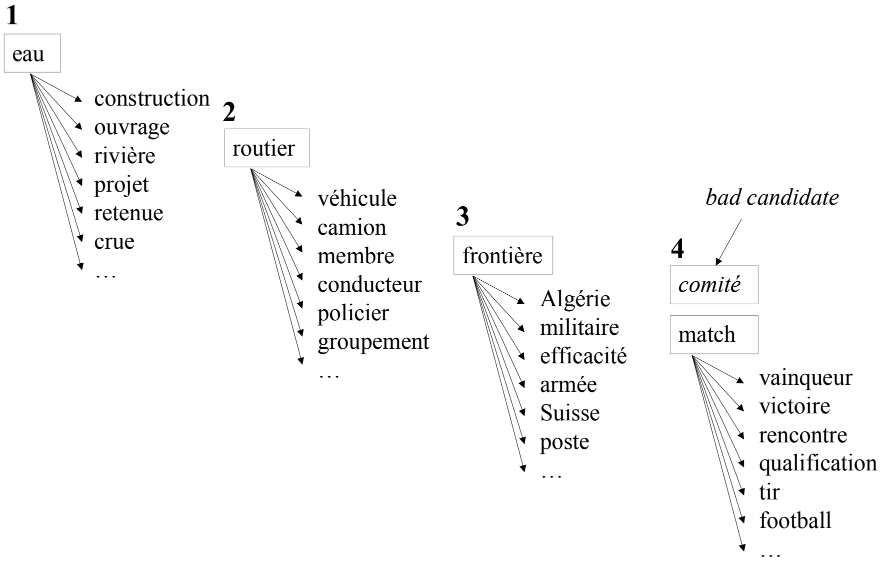

Figure 7. Step-by-step deletion of neighbors

When the graph's smallest components are reached, it becomes highly likely that too many nodes have been deleted, and some corrective heuristics are needed to make sure the next node examined is in fact a good candidate for the role of root hub. The node has to have (1) at least 6 specific neighbors (this threshold was determined experimentally), and (2) a weighted clustering coefficient large enough for it to actually be a root hub of a bundle.

For the sake of rapidity, a rough approximation that turned out to be sufficient is used for criterion 2: the mean of the weights between the candidate node and its 6 most frequent neighbors must simply be below a given threshold (set at 0.8 based on experimentation). Likewise, instead of going through the nodes in decreasing order of degree, which requires computations at the node level, they are scanned in decreasing order of *frequency* (this information was already computed during context preprocessing); given that these two values are highly correlated (see above), the result is nearly identical.

Upon completion of the first step, we have the root hub of each component. Each use is now characterized by its root hub and its most frequent specific neighbors.

Taking the *barrage* example once again, the four components can be characterized as follows:

EAU	*construction ouvrage rivière projet retenue crue* (construction engineering-work river project reservoir flood)
ROUTIER	*véhicule camion membre conducteur policier groupement* (vehicle truck member driver policeman group)
FRONTIERE	*Algérie militaire efficacité armée Suisse poste* (Algeria military efficiency army Switzerland post)
MATCH	*vainqueur victoire rencontre qualification tir football* (winner victory encounter qualification shot soccer)

This information may be sufficient for helping users narrow down their queries. However, it does not delineate the exact composition of the component (this is done in the second step of the algorithm).

More formally, the algorithm can be written as follows:

```
RootHubs($G$, $Freq$) {
    $G$ : cooccurrence graph
    $Freq$ : array of frequencies of nodes in $G$

    $V \leftarrow$ array of nodes in $G$ sorted in decreasing order of frequency
    $H \leftarrow \emptyset$

    while $V \neq \emptyset$ et $Freq(V[0]) > threshold$ {
        $v \leftarrow V[0]$
        if GoodCandidate($v$)
        then {
            $H \leftarrow H \cup v$
            $V \leftarrow V - (v \cup \Gamma(v))$
        }
    }
    return $H$
}
```

The algorithm is very fast, since once the nodes are sorted in decreasing order of frequency (which is in fact performed during corpus preparation), it operates in *O*(*N*), where *N* is the number of graph nodes (the number of neighbor-deletion operations is at most equal to *N*).

4.2. Delineating Components

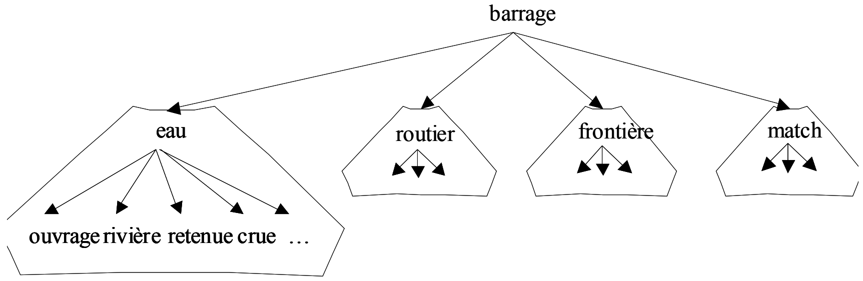

Figure 8. Minimum spanning tree and high-density components

Delineating the high-density components amounts to attaching each node to the root hub closest to it. Because of the graph's small world structure, where all nodes are close to each other in terms of the number of edges to cross, the edge weights can be put to good use: the distance between two nodes will be measured by the smallest sum of the weights of the edges on the paths linking them. After adding the target word (which is not in the cooccurrence graph -- here, *barrage*), a *minimum spanning tree* (or MST) is computed over the graph by taking the target word as the root and making its first level consist of the previously identified root hubs.[12] The components correspond to the main branches of the tree (Figure 8).

The algorithm is as follows:

```
Components($G, H, t$) {
    $G$ : cooccurrence graph
    $H$ : set of root hubs
    $t$ : target word

    $G' \leftarrow G \cup t$
    for each $h$ in $H$ {
        add edge $<t, h>$ with weight 0 to $G'$
    }

    $T \leftarrow$ MST($G', t$)

    return $T$
}
```

The algorithm's complexity is no greater than that of the minimum spanning tree computation, which can be efficiently performed by Kruskal's (1956) algorithm, well suited to sparse graphs. In the worst of all cases, its complexity is $O(E \lg E)$, where E is the number of edges in the graph. However, Kruskal's algorithm is known to behave in a quasi-linear fashion in most concrete cases.

5. Disambiguation

The minimum spanning tree can be used to easily construct a disambiguator for tagging target word occurrences in the corpus. Each tree node v is assigned a score vector **s** with as many dimensions as there are components:

$$\mathbf{s}_i = \frac{1}{1+d(h_i, v)} \text{ if } v \text{ belongs to component } i \qquad (12)$$

$$\mathbf{s}_i = 0 \text{ otherwise.} \qquad (13)$$

In (11), $d(h_i, v)$ is the distance between root hub h_i and node v in the tree.

Formula (11) assigns a score of 1 to root hubs, whose distance from themselves is 0. The score gradually approaches 0 as the nodes move away from their root hub. For example, *pluie* (rain) belongs to the component *EAU* (water) and *d*(*eau*, *pluie*) = 0.82; its score vector is (0.55 0 0 0). Likewise, *saison* (season) belongs to the component *MATCH* and *d*(*match*, *saison*) = 1.54; its score vector is (0 0 0 0.39).

For a given context, the score vectors of all words in that context are added, and the component that receives the heaviest weight is chosen. For example, the scores for the context:

> *Le* ***barrage*** *recueille l'eau à la saison des pluies.*
> (The **dam** collects water during the rainy season)

are shown in Table 4. *EAU* is the component with the highest total score (1.55), followed by *MATCH* (0.39).

[12] A minimum spanning tree is a tree that goes through all nodes of the initial graph while minimizing the sum of the edge weights. Various standard algorithms are available for efficiently computing minimum spanning trees.

	EAU	ROUTIER	FRONTIERE	MATCH
S_{eau}	1.00	0.00	0.00	0.00
S_{saison}	0.00	0.00	0.00	0.39
S_{pluie}	0.55	0.00	0.00	0.00
Total	**1.55**	**0.00**	**0.00**	**0.39**

Table 4. Scores for the context "Le barrage recueille l'eau à la saison des pluies"

A reliability coefficient varying between 0 and 1 can be calculated from the difference, Δ, between the best score and the second best score:

$$\rho = 1 - \frac{1}{1+\Delta} \tag{14}$$

In the preceding example, the reliability coefficient ρ is 1 - (1 / (1 + 1.55 - 0.39) = 0.54.

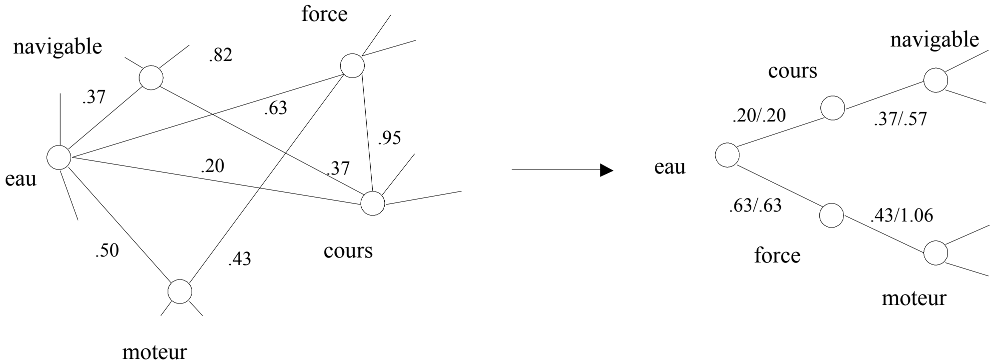

Figure 9. Graph expansion

It is interesting to take a closer look at what the algorithm is actually doing from the linguistic standpoint. Let us take the real example shown in Figure 9. On the left side of the figure, one can see that the words *eau, navigable, force, cours, moteur* (water, navigable, power, course, motor) are highly interconnected (5 connections out of a possible 6). Yet not all of these connections are alike. The relations *eau - cours, cours - navigable, eau - force,* and *force - moteur* (water - course, course - navigable, water - power, and power - motor) are primary relations, which appear in expressions like *cours d'eau, cours navigable, force de l'eau,* and *force motrice* (watercourse, navigable course, water power, motor power). The other relations are secondary, which occur by means of transitivity following a principle of the type "The friends of my friends also become my friends". There is no particular relation between water and motor except by way of water power, which is motor-based. Computing the minimum spanning tree "de-clusters" the graph by pointing out primary relations between words. The algorithm thus "expands" the small lexical world by showing us which relations are preferential: among the 4 neighbors of*eau* in this example, only two are still its neighbors in the final tree.

6. Viewing and Navigation

Difficult viewing problems arise for large graphs, given that most drawings are NP-hard. We use a method recently devised by Munzner (2000), which allows a very fast display using hyperbolic trees.[13] We feed the algorithm with the MST previously described, adjusted by a number or heuristics.

[13] We use Munzner's graphic library *H3Viewer*: http://graphics.stanford.edu/~munzner/h3/

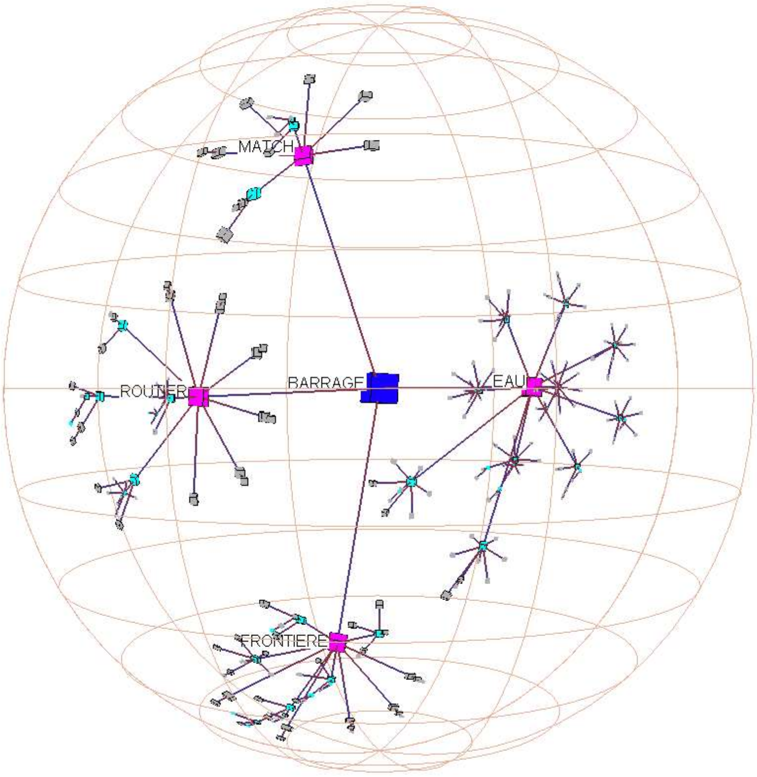

Figure 10. *Barrage* : main view

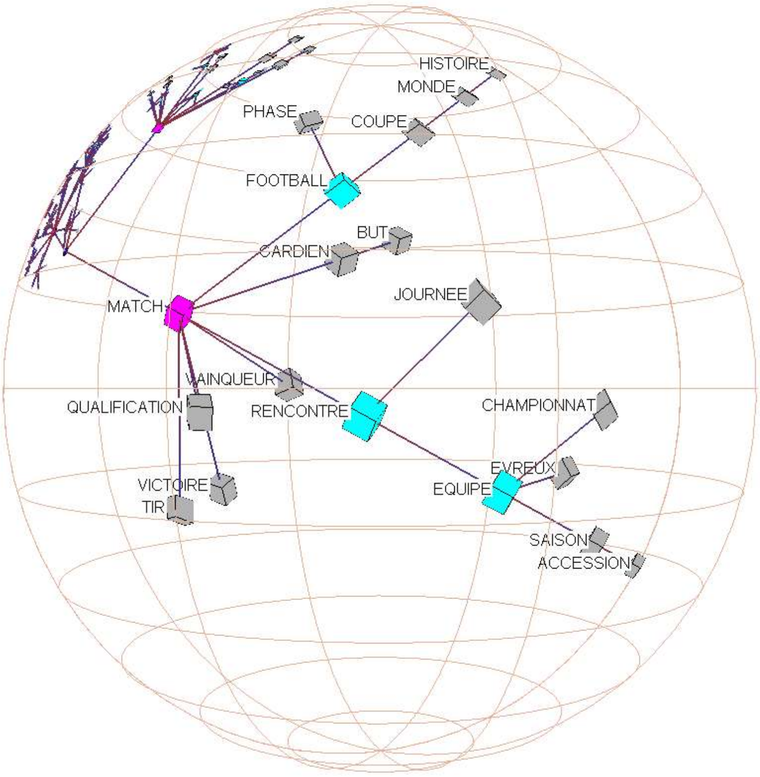

Figure 11. *Barrage* : view centered on the root hub *match*

Figure 10 shows the main view (top of the tree) for the word *barrage*. The user can navigate from domain to domain inside the hyperbolic drawing using the mouse: a left click on a node centers the display on that node; a left drag moves a node and changes the context; a right drag rotates the tree. Figure 11 shows the view obtained by clicking on the node match. An addition will be made in future implementations so that the user can view the corpus contexts closest to each node of the tree.

While navigating through the graph, one can also explore secondary domains within a given component. Figure 12 illustrates a subdomain of the hydraulic dam use of barrage, obtained by clicking on *construction* (construction) and then *coût* (cost).

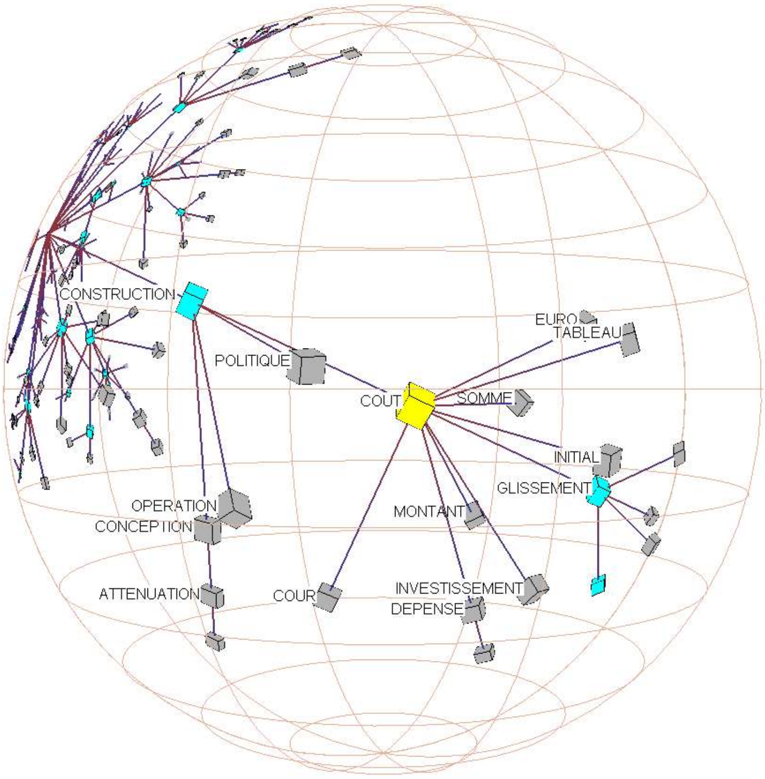

Figure 12. *Barrage : construction → coût* subdomain

Finally, the program can display all edges of the graph, i.e., its transversal links in addition to the ones in the backbone tree. This type of display highlights the division into within-component links and between-component links. In Figure 13, for example, which shows the transversal links for the target word barrage, we can see that between-component edges are scarce, which means that the uses are clearly distinct. Inversely, Figure 14 for the word *vol* (flight, gliding, theft, ...) points out numerous transversal links between the components *LIBRE* (free) and *AVION* (airplane). In navigating through the graph and looking more closely at the reasons behind these relations, we find out that these two components share an important subdomain, that of vacation: *loisirs, soleil*, ... (leisure, sun, ...). By contrast, the component (*VOL A) VOILE* (gliding) is weakly connected to the vacation subdomain (at least in this corpus): the concerned pages are mostly technical.

HyperLex thus seems to supply a useful tool for domain and lexicon navigation. It remains to be seen whether its utilization by the general public is a realistic possibility, but it does indeed appear to provide a valuable means of exploration for terminologists and lexicographers.

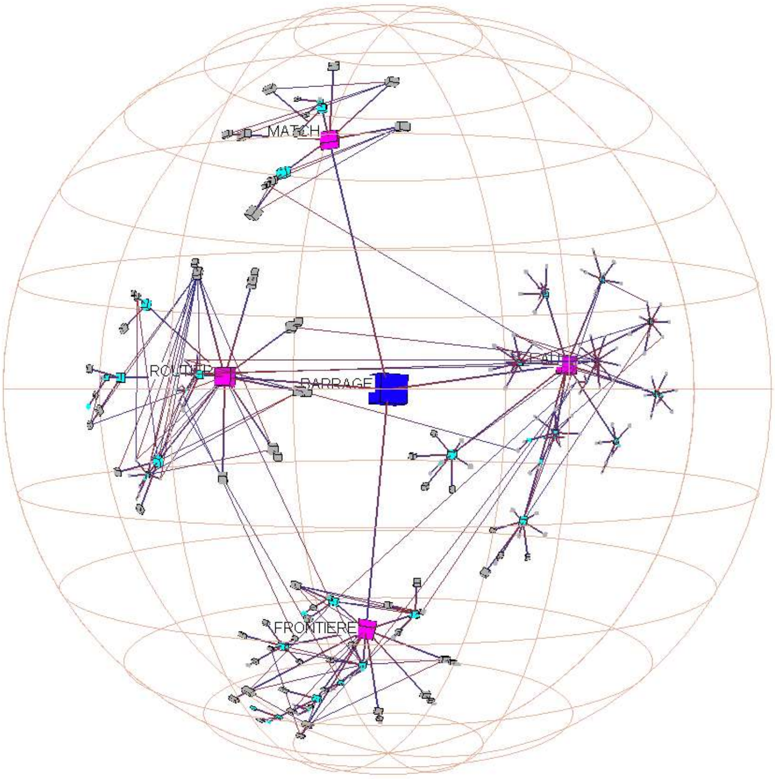

Figure 13. *Barrage* : complete representation of the graph

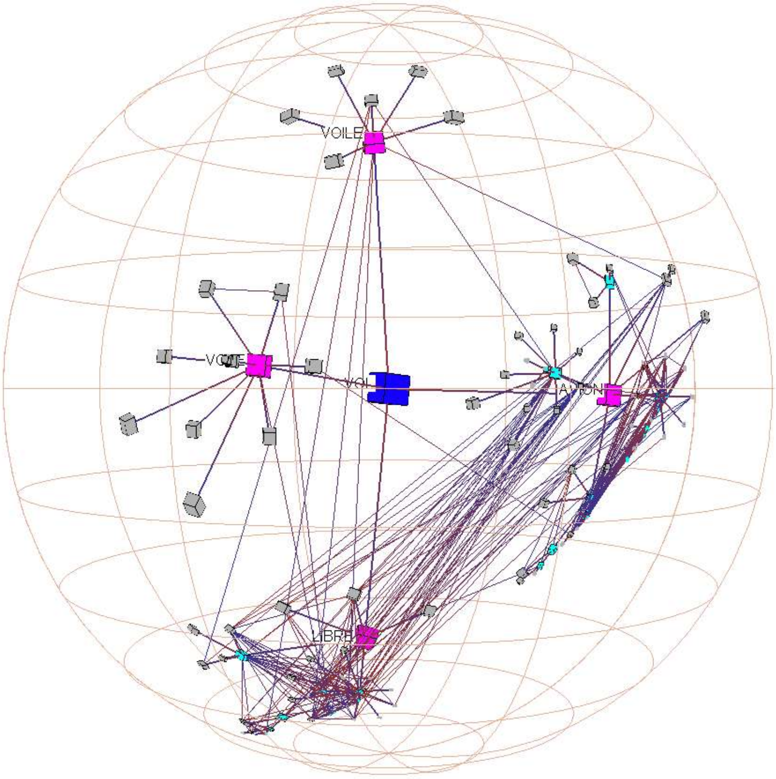

Figure 14. *Vol* : links between components (*vol libre* ↔ *avion*)

7. Evaluation

The results of the *HyperLex* algorithm were evaluated on the Web page corpus, using the list of ten test words described above.

The first step was to make sure that the algorithm correctly extracted most uses in the corpus, irrespective of any tagging of the contexts. This subtask is interesting in its own right (for proposing query refinements to users, for example).

The next step was to assess the quality of the context tagging by drawing a random sample from the corpus and taking the standard recall and precision measures used in word sense disambiguation. However, these measures cannot accurately assess an algorithm's efficiency on infrequent uses; they can only reflect behaviour on the principal use. To make up for this, a third, much more stringent measure was added: the precision level achieved on the 25 best contexts returned by the algorithm for *each* use discriminated.

No attempt was made at evaluating the behaviour of the algorithm on query terms. Spink et al. (2001) report that the average length of Web queries is 2.4 words, which deviates considerably from traditionnal IR (TREC-like) searches. In addition, close examination of major search engine information sites shows that the most popular queries are multi-token proper names or titles such as *Britney Spears, Harry Potter, American Airlines, Sex and the City,* or *Lord of the Ring*. Real multi-term queries (such as *gay marriage* or *Oscar nominations*) do exists, but they are far less common (we are not aware of studies reporting precise figures). The real query length is probably much closer to one word and disambiguation techniques are probably bound to failure in the general case. A more promising strategy seems to consists in a presentation of results categorised according to word uses, or a "Refine your results" box as can be seen on the current versions of *Excite* or *Altavista*.

7.1. List of Uses

Table 6 gives the list of uses extracted by *HyperLex* from the entire corpus. A total of 50 uses were identified, making for an average of 5 per target word. To check for the completeness and relevance of the list, 100 contexts were drawn at random for each test word, making 1000 contexts in all.

The system made two kinds of omissions. Some pertained to general uses of the target word, i.e., uses that are not related to a particular domain. For instance, for the word *barrage*, the use *faire barrage à* (stand in the way of) was not detected. Only four instances of this expression were found in the context sample tested. In other words, it was about as frequent as *match de barrage*, so quantitatively speaking, it could have been detected. It just so happens, though, that *faire barrage à* is a general expression in French and is found in a wide variety of contexts: *faire barrage à un projet, à quelqu'un* (stand in the way of a plan, of someone), etc. Another typical example is the word *solution*, for which the algorithm did not detect the general sense of "solution to a problem", a very frequent use (16 occurrences here) but one that can appear in any domain.

All in all, 8 general uses were missing for the entire set of 10 target words, and only 3 of them had a frequency above 5 (Table 5). These omissions are not very troublesome, since they do not concern specific uses, and in my mind, would hardly ever be put in a query. Besides, they only concerned 6.6% of the contexts.

	All		**Frequency >5**	
	Uses	Contexts	Uses	Contexts
GENERAL	8	6,6%	3	5,1%
SPECIFIC	36	7,4%	1	1,9%
Total	44	14,0%	4	7,0%

Table 5. Omissions

Omissions of specific uses are more troublesome because they would automatically lead to query failure. Note, however, that few such omissions were made here: although a total of 36 specific uses were omitted (Table 5), 19 had a single occurrence in the test sample, and only one had a frequency of 5 or more. The latter was one of the uses of the word *lancement* (launching), "lancement d'un programme" (launching of a program), which occurred 19 times in the random sample of 100 contexts, which is a high frequency. A closer look at the contexts showed that the themes for this use were scattered (without its being a "general" use in the sense stated above). The word programme itself was rarely present in the context, and it was usually a question of explanations about running (launching) such and such a command or application, in extremely varied domains. It is not surprising, then, that the algorithm failed to isolate this use, despite its importance in the language.

Target	Root	Most frequent neighbors	Freq (%)[14]
BARRAGE	EAU	construction ouvrage rivière projet retenue crue	67-85
	ROUTIER	véhicule camion membre conducteur policier groupement	3-16
	FRONTIERE	Algérie militaire efficacité armée Suisse poste	5-19
	MATCH	vainqueur victoire rencontre qualification tir football	1-10
DETENTION	PROVISOIRE	juge liberté loi procédure prison instruction	78-93
	DETENU	police centre autorité arrestation torture arbitraire	4-18
	ARME	autorisation acquisition feu munition vente commerce	0-8
	ANIMAL	transport compagnie sauvage certificat annexe directive	0-6
FORMATION	PROFESSIONNEL	centre entreprise organisme stage service programme	96-100
LANCEMENT	SATELLITE	Ariane programme spatial lanceur orbite fusée	94-100
	PRODUIT	public entreprise événement convention presse affaire	0-6
ORGANE	DON	transplantation greffe donneur prélèvement tissu vie	30-51
	DELIBERANT	public établissement président demande attribution communauté	8-24
	REGLEMENT	pays appel différend OMC réunion autorité	12-30
	TECHNIQUE	scientifique convention économique conférence subsidiaire programme	0-8
	CONSULTATIF	matière civil tête supervision mémorandum PAB	1-12
	MALADIE	cœur traitement spécimen preuve sang intervention	0-4
	REPRESENTANT	délégué suprême concertation département personnel agent	0-9
	PARTI	presse chef journal Genève Allemagne rédacteur	0-9
PASSAGE	EURO	public travail entreprise système national monnaie	41-63
	AN_2000	programme autorité installation réseau solution matériel	2-13
	NIVEAU	porte chemin ouverture salle route entrée	0-9
	LIBRE	cour prestation police assurance caisse prévoyance	3-16
	CHEVAL	main énergie équilibre trot dos foulée	0-4
	PARAMETRE	mode appel variable argument langage expression	2-14
	GALERIE	ville boutique bois panorama époque verrière	0-8
	TERRE	durée mouvement soleil Vénus Mercure nœud	4-18
	MORT	rite Dieu naissance Christ vivant Jésus	0-4
RESTAURATION	HOTELLERIE	formation durée centre professionnel entreprise alternance	23-43
	CONSERVATION	sauvegarde atelier monument technique historique oeuvre	34-55
	HEBERGEMENT	activité hôtel région loisir culture contact	0-8
	RAPIDE	restaurant vente établissement repas marche traiteur	1-10
	FICHIER	système information donnée client espace bande	7-23
	PIERRE	bâtiment chantier terre polychromie taille sec	0-4
	MEUBLE	bois table mobilier décoration fabrication antiquité	1-10
SOLUTION	GESTION	entreprise service logiciel client information système	75-91
	JEU	monde gratuit astuce joueur gain francophone	0-4
	INJECTABLE	perfusion glucose HOP commercialisation arrêt Fandre	7-23
STATION	SKI	hiver piste montagne sport village location	75-91
	METEO	température Oregon scientific WS professionnel capteur	0-6
	SPATIAL	international MIR système programme ISS projet	4-18
	TRAVAIL	réseau traitement donnée carte Sun environnement	1-10
	RADIO	navire région réception installation antenne communication	0-6
	PRIMAGAZ	Paris aire Esso province Marseille Dyneff	0-4
	EAU	épuration source mer plage Yves rivière	0-4
	LIGNE	métro quai terminus voyageur correspondance atelier	0-4
VOL	AVION	billet pilote club sec départ voyage	51-72
	LIBRE	école parapente loisir montagne formation Paris	23-43
	VOILE	centre photo vent pilotage forum compétition	0-4
	VOLÉ	service recherche numéro base donnée véhicule	2-13

Table 6. Main uses of the test words

[14] 95% confidence interval computed using the binomial law.

In summary, the behavior of *HyperLex* in terms of its ability to detect IR-relevant uses can be considered quite good from a quantitative point of view: one can legitimately claim that nearly all uses whose frequency was above 5% were well detected.

Qualitatively speaking, the proposed use division was adequate in most cases. Certain use divisions emerged that would probably not be proposed by a lexicographer and therefore might seem surprising at first glance. A case in point is the distinction made between two of the uses of the word detention, identified by the root hubs *DETENU* (prisoner) and *PROVISOIRE* (provisional). In both cases, persons were in prison, as opposed to the other uses detected (detention of arms, animals). However, a more careful look at the pages in question showed that the subcorpus contained two clearly disjoint domains: one (the *DETENU* hub) pertained to the human aspects of imprisonment (conditions of detention, torture, visits, etc.), the other (the *PROVISOIRE* hub), to its legal aspects (remand in custody, laws, etc.). In an information retrieval perspective, it is not illogical to distinguish the two domains, even though it might be useful to hierarchically group some of the uses. The extent of between-component linkage would perhaps be a good index for doing so. This topic deserves further research.

Inversely, for the word *station,* the algorithm merged public radio stations (FM, etc.) and radio stations on ships, two lexically close fields (radio, communication, MHz, etc.). Yet one would certainly want to separate these two uses, since they would most likely be used in different queries. This suggests a possible enhancement of the algorithm: taking the distance between cooccurrents into account. For instance, the expression *station de radio* (radio station) is used only for public radio stations, while the expression *station de navire* (ship station) is found in maritime contexts, with the word radio itself usually being found farther away in the context as part of other expressions like *opérateur radio* (radio operator) and *équipement radio* (radio equipment), etc.

Finally, in some cases, it is the description by the root hub and its neighbors that is inadequate. There are various reasons for this. Take the example of one of the uses of *station*, which was identified by the hub *PRIMAGAZ*, a brand of LPG gas. The reason is that the Internet contains many lists of gas stations that sell LPG, apparently because obtaining gas is a major concern for people with vehicles that run on this type of fuel. LPG would be a more appropriate root hub, but while most of the concerned pages had LPG in the heading, the paragraphs containing the word station contained only the various brand names (*Primagaz, Shell, Esso,* etc.) and addresses. A more global page analysis would perhaps be useful, at least for labelling uses.

7.2. Global Tagging

The disambiguator described in Section 5 was used on the 10 test-word subcorpora. When several contexts contained the target test word on the same Web page, the most reliable use (assessed by the reliability coefficient ρ) was applied to all contexts on the page. This coefficient can also serve as a control for tagging recall. The value chosen was $\rho \geq 0.5$, which corresponds to a difference of 1 between the best two scores. This gave a recall rate of 82%, which is more than sufficient for the application at hand.

For each target word, 100 contexts were drawn at random (1000 in all) from among those with a $\rho \geq 0.5$, and the suitability of the use proposed by the algorithm was verified by hand. Verification by several experts would have been preferable, but, given the cost of the task, we relied on a single expert judgement, a strategy which is rather standard in the field. The appropriateness of word uses in a given context was judged on the basis of the list of frequent neighbours provided by the algorithm, which can be seen as a sort of "dictionary definition". For example, the use *EAU* of *barrage* is characterised by {*eau, construction, ouvrage, rivière, projet, retenue, crue*} (water, construction, engineering work, river, project, reservoir, flood), whereas the use *ROUTIER* is characterised by {*routier*, *véhicule, camion, membre, conducteur, policier, groupement*} (vehicle, truck, member, driver, policeman, group).

For each subcorpus, tagging precision was checked, along with the baseline obtained by taking the most frequent word use and assigning it to each instance in the collection (Table 7).

Keeping the 73% baseline precision rate in mind, one can see that the overall precision was 97%,[15] which is excellent. An error reduction measure error, *ER*, can be used to judge exactly how good the algorithm's work is:

$$ER = \frac{precision - baseline}{1 - baseline}$$

HyperLex reduced the error by 90.4%[16] compared to what one would obtain by trivial tagging in terms of the most frequent use (this figure does not include the word *formation*, which posed no problems because its subcorpus had only one use). The *ER* measure clearly highlights words that are more difficult than others, here, *organe* (organ) and *passage* (passage) and to some extent, *solution* (solution). Except for these three words, the algorithm performed error-free tagging.

Test word	Precision	Baseline	Error reduc.
BARRAGE	1.00	0.77	100.0%
DETENTION	1.00	0.87	100.0%
FORMATION	1.00	1.00	n/a
LANCEMENT	1.00	0.99	100.0%
ORGANE	0.88	0.40	80.0%
PASSAGE	0.88	0.52	75.0%
RESTAURATION	1.00	0.44	100.0%
SOLUTION	0.98	0.84	87.5%
STATION	1.00	0.84	100.0%
VOL	1.00	0.62	100.0%
Total	**0.97**	**0.73**	**90.4%**

Table 7. Tagging precision

The frequencies of the different uses in the corpus were estimated based on the manual tagging used as a reference. The estimates are reported in the *Freq* column of Table 6.

7.3. Most Relevant Pages

As mentioned at the beginning of this section, the conventional measure of precision mainly reflects the algorithm's behavior on the principal use. Precision was therefore assessed for each test-word use by looking at the best 25 contexts (in terms of the reliability coefficient ρ). This measure is much more severe than the preceding one because it weights each use equally, even the rarest ones. It is nevertheless quite realistic, in that it corresponds to the behavior of search engines that categorize the results before presenting them to the user. The choice of 25 pages was deemed sufficient based on Silverstein, Henzinger, Marais, and Moricz's (1999) study on 150 million *Altavista* queries: in 85.2% of the queries, only the first screen page of 10 results was examined, with an overall average of 1.39 screens.

Test word	Contexts	Precision
BARRAGE	100	1.00
DETENTION	100	1.00
FORMATION	25	1.00
LANCEMENT	50	1.00
ORGANE	195	0.86
PASSAGE	225	0.92
RESTAURATION	175	0.94
SOLUTION	75	1.00
STATION	200	1.00
VOL	100	1.00
Total	**1245**	**0.96**

Table 8. Precision on 25 first pages per use

[15] 95% confidence interval computed using the binomial law : $CI_{95\%}$ = 96-98%.

[16] $CI_{95\%}$ = 86-94%

The best 25 contexts were checked for each of the 50 uses in Table 7, that is, 1245 contexts in all.[17] The overall precision was 95.5% (Table 8).[18] Once again, a few errors cropped up on three words (*organe* and *passage* as before, and *restauration*). Apart from these three words, all contexts returned did in fact pertain to the proper use, which is appreciable, since quite a few of the detected uses had an estimated frequency under 5%.

8. Conclusion

This article presented an efficient algorithm for disambiguating the senses of words in information retrieval tasks. The algorithm, *HyperLex*, makes use of the particular structure of cooccurrence graphs, shown here to be "small worlds", a topic of extensive research in recent years. As in previously proposed methods (word vectors), the algorithm automatically extracts a use list for the words in a corpus (here, the Web), a feature that sidesteps problems brought about by recourse to a preestablished dictionary. However, unlike earlier methods, *HyperLex* can detect low-frequency uses (as low as 1%). An evaluation on 10 highly polysemous test words showed that the system detected a great majority of the relevant uses. The system's in-context word-tagging method proved remarkably precise and therefore can offer high-quality categorization of query output. Enhancements are of course possible, but this study already seems to cast doubt on the idea that disambiguation techniques are useless if not detrimental in IR. The excellent results obtained here seem to constitute an important step forward in word sense disambiguation, one that goes beyond IR applications alone.

Finally, *HyperLex* is associated with a viewing and navigation technique that allows the user to navigate in the lexicon and domains of a corpus. Its utilization by the general public has yet to be tested, but the tool already appears to be a useful instrument for terminologists, lexicographers, and other specialized users.

9. Epilogue

After this study was conducted, I discovered, owing to the (impressive) review by Albert and Barabási (2002) that other researchers have been independently investigating the possibility of modelling the semantic aspects of human language using small world networks. Albert and Barabási cite a study in their own group, by Yook, Jeong and Barabasi, which shows that the network of synonyms extracted from the Merriam-Webster Dictionary exhibits a small-world and scale-free structure (however this study has never been published, and is not available on the Web[19]). In another study, Ferrer i Cancho and Solé (2001) have constructed a network of words that co-occur in sentences in the British National Corpus (at a maximum distance of two), and they again found a similar structure for the resulting network. This seems to provide independent confirmation that associative lexical relations follow the power-law, scale-free pattern that we have reported here. Associative networks have received a lot of attention from psychologists since Collins and Quillian (1969). In a recent, unpublished paper, Steyvers and Tenebaum (2004) indicate that networks built from a large human free word-association database, Roget's thesaurus and Wordnet all share the distinctive features of small-world and scale-free structures, a fact which seems to open new avenues of research on semantic memory models. Interestingly, they arrive at the same pessimistic conclusions about the capacity of Euclidian spaces to represent adequately words senses and uses, whose frequency seem governed by power laws.

[17] One of the uses of *organe* had only 20 contexts.

[18] $CI_{95\%}$ = 94.2 - 96.6%.

[19] Albert-László Barabási, personal communication (March 2004)

Acknowledgements

The author is grateful to Vivian Waltz for her help with the translation, and to anonymous reviewers for their very useful comments.

References

Ahlswede T. E. 1993. Sense Disambiguation Strategies for Humans and Machines. In: Proceedings of the 9th Annual Conference on the New Oxford English Dictionary, Oxford (England), pp. 75-88.

Ahlswede T. E. 1995. Word Sense Disambiguation by Human Informants. In: Proceedings of the Sixth Midwest Artificial Intelligence and Cognitive Society Conference, Carbondale (Illinois), pp. 73-78.

Ahlswede T. E., Lorand D. 1993. The Ambiguity Questionnaire: A Study of Lexical Disambiguation by Human Informants. In: Proceedings of the Fifth Midwest Artificial Intelligence and Cognitive Society Conference, Chesterton (Indiana), pp. 21-25.

Albert, R., Barabási. A.-L. 2002. Statistical mechanics of complex networks. Review of Modern Physics 74:47-97.

Amsler R. A., White J. S. 1979. Development of a computational methodology for deriving natural language semantic structures via analysis of machine-readable dictionaries. Final report on NSF project MCS77-01315. University of Texas at Austin, Austin (Texas).

Barabási, A.-L. , Albert, R. 1999. Emergence of scaling in random networks. Science 286:509–512.

Bruce, R., Wiebe, J. 1998. Word sense distinguishability and inter-coder agreement. In: Proceedings of the 3rd Conference on Empirical Methods in Natural Language Processing (EMNLP-98). Association for Computational Linguistics SIGDAT, Granada (Spain), pp. 53-60.

Collins, A.M., Quillian, M.R. 1969. Retrieval time from semantic memory. Journal of Verbal Learning and Verbal Behaviour 8:240-247.

Fellbaum, C., Grabowski, J., Landes, S. 1998. Performance and confidence in a semantic annotation task. In: Fellbaum, C. (ed.) WordNet:An electronic database. Cambridge (Massachusetts). The MIT Press, pp 217-237.

Harris, Z. S. 1954. Distributional Structure. Word 10:146-162.

Ferrer i Cancho, R., Solé, R. V. 2001. The small world of human language. Proceedings of The Royal Society of London. Series B, Biological Sciences, 268(1482):2261--2265.

Gelbukh, A; Sidorov, G; Chanona-Hernandez, L. 2003 Is Word Sense Disambiguation Useful in Information Retrieval? Paper given at the first International Conference on Advances in Infrastructure for e-Business, e-Education, e-Science, e-Medicine, and Mobile Technologies on the Internet, L'Aquila, Italy.

Hornby, A. S. 1954. A guide to patterns and usage in English. London, OUP.

Hornby, A.S., Gatenby, E.V., Wakefield, H. 1942. Idiomatic and Syntactic English Dictionary. [Photographically reprinted and published as A Learner's Dictionary of Current English by Oxford University Press, 1948; subsequently, in 1952, retitled The Advanced Learner's Dictionary of Current English.] Kaitakusha, Tokyo.

Ide, N. M., Véronis, J. 1998. Introduction to the special issue on word sense disambiguation:the state of the art. Computational Linguistics 24(1):1-40.

Jansen, B. J., Spink, A., Saracevic, T. 2000. Real life, real users, and real needs: A study and analysis of user queries on the web. Information Processing and Management 36(2):207-227.

Jorgensen, J. 1990. The psychological reality of word senses. Journal of Psycholinguistic Research 19:167-190.

Kilgarriff A 1998 SENSEVAL:An Exercise in Evaluating Word Sense Disambiguation Programs. In Proceedings of the Language Resources and Evaluation Conference. Granada (Spain), pp 581-588.

Krovetz, R, Croft, W. B. 1992. Lexical Ambiguity and Information Retrieval. ACM Transactions on Information Systems 10(2):115-141.

Kruskal, J. B. 1956. On the shortest spanning subtree of a graph and the traveling salesman problem. In: Proceedings of the American Mathematical Society, volume 7, pp. 48-50.

Meillet, A. 1926. Linguistique historique et linguistique générale. Vol. 1. Champion, Paris, 351pp. (2nd edition).

Milgram, S. 1967. The small world problem. Psychology Today 2:60–67.

Munzner, T. 2000. Interactive visualization of large graphs and networks. Ph. D. Dissertation, Stanford University. Stanford (California).

Newman, M. E. J. 2003. The structure and function of complex networks. SIAM Review 45:167-256.

Reymond, D. 2002. Dictionnaires distributionnels et étiquetage lexical de corpus. In: Actes de TALN/RECITAL'2001, Atala, Tours (France), pp. 479-488.

Salton, G., McGill, M. 1983. Introduction to Modern Information Retrieval. McGraw-Hill, New York.

Sanderson, M. 1994. Word sense disambiguation and information retrieval. In: Proceedings of the 17th ACM SIGIR Conference, Dublin (Ireland), pp. 142-151.

Schütze, H. 1998. Automatic word sense discrimination. Computational Linguistics 24(1):97-124.

Schütze, H. Pedersen, J. 1995. Information retrieval based on word senses. In: Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval, Las Vegas (Nevada), pp. 161-175.

Silverstein, C., Henzinger, M., Marais, H., Moricz, M. 1999. Analysis of a Very Large AltaVista Query Log. SRC Technical note #1998-14. [On-line at http://gatekeeper.dec.com/pub/DEC/SRC/technical-notes/abstracts/src-tn-1998-014.html]

Spink, A., Wolfram, D., Jansen, B. J., Saracevic, T. 2001. Searching of the web: the public and their queries. Journal of the American Society of Information Science and Technology 52(3):226 - 234.

Steyvers, M., Tenenbaum, J. B. 2004. The large-scale structure of semantic networks: statistical analyses and a model of semantic growth. Unpublished. [Online at http://psiexp.ss.uci.edu/research/papers/small9formatted.pdf].

Stevenson, M., Wilks, Y. 2001. The interaction of knowledge sources in word sense disambiguation. Computational Linguistics 27(3):321-349.

Stokoe, C; Oakes, P. M; Tait, J. 2003. Word sense disambiguation in information retrieval revisited. In: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval Toronto, Canada, pp. 159-166.

Thorndike, E. L., Lorge, I. 1938. Semantic counts of English Words, Columbia University Press, New York.

Véronis, J. 1998. A study of polysemy judgements and inter-annotator agreement, In: Programme and advanced papers of the Senseval workshop, Herstmonceux Castle (England), pp. 2-4. [Online at http:/www.up.univ-mrs.fr/veronis/pdf/1998senseval.pdf].

Véronis, J. 2001. Sense tagging: does it make sense? Paper presented at the Corpus Linguistics'2001 Conference, Lancaster, U.K. [Online at http://www/up.univ-mrs.fr/veronis/pdf/2001-lancaster-sense.pdf]

Véronis, J. 2003. Hyperlex : cartographie lexicale pour la recherche d'informations. In: Actes de la Conférence Traitement Automatique des Langues (TALN'2003), Batz-sur-mer (France), pp. 265-274. [Online at http://www.up.univ-mrs.fr/veronis/pdf/2003-taln.pdf]

Voorhees E. M. 1993. Using WordNet to disambiguate word sense for text retrieval. In: Proceedings of ACM SIGIR Conference, pp. 171-180.

Wallis P. C. 1993. Information retrieval based on paraphrase. In: Proceedings of the First Pacific Association for Computational Linguistics Conference, PACLING Conference, Vancouver (Canada).

Watts, J.W., Strogatz, S.H. 1998. Nature 393:440-442.

Weiss, S.F. 1973. Learning to disambiguate. Information Storage and Retrieval 9:33-41.

Wittgenstein, L. 1953. Philosophische Untersuchungen [Philosophical Investigations, translated by G.E.M. Anscombe, New York, Macmillan.]