Lexical Access Based on Underspecified Input

Michael ZOCK
LIF-CNRS
Équipe TALEP
163, Avenue de Luminy
F-13288 Marseille Cedex 9
michael.zock@lif.univ-mrs.fr

Didier SCHWAB
Groupe GETALP
Laboratoire d'Informatique de Grenoble
385 avenue de la Bibliothque - BP 53
F-38041 Grenoble Cedex 9
didier.schwab@imag.fr

Abstract

Words play a major role in language production, hence finding them is of vital importance, be it for speaking or writing. Words are stored in a dictionary, and the general belief holds, the bigger the better. Yet, to be truly useful the resource should contain not only many entries and a lot of information concerning each one of them, but also adequate means to reveal the stored information. Information access depends crucially on the organization of the data (words) and on the navigational tools. It also depends on the grouping, ranking and indexing of the data, a factor too often overlooked.

We will present here some preliminary results, showing how an existing electronic dictionary could be enhanced to support language producers to find the word they are looking for. To this end we have started to build a corpus-based *association matrix*, composed of *target words* and *access keys* (meaning elements, related concepts/words), the two being connected at their intersection in terms of *weight* and *type of link*, information used subsequently for grouping, ranking and navigation.

1 Context and problem

When speaking or writing we encounter basically either of the following two situations: one where everything works automatically, somehow like magic, words popping up one after another like spring water, and another where we look deliberately and often painstakingly for a specific, possibly known word. We will be concerned here with this latter situation: a speaker/ writer using an electronic dictionary to look for such a word. Unfortunately, alphabetically organized dictionaries are not well suited for this kind of *reverse lookup* where the inputs are meanings (elements of the word's definition) or conceptually related elements (collocations, associations), and the outputs the target words.

Without any doubt, lexicographers have made considerable efforts to assist language users, building huge resources, composed of many words and lots of information associated with each one of them. Still, it is not unfair to say most dictionaries have been conceived from the reader's point of view. The lexicographers have hardly taken into account the language producer's perspective,[1] considering conceptual input, incomplete as it may be, as starting point. While *readers* start with words, looking generally for their corresponding meanings, *speakers* or *writers* usually start with the opposite, meanings or concepts,[2] which should be the entry points of a dictionary, which ideally is neutral in terms of access direction.[3]

The problem is that we still don't know very well what *concepts* are, whether they are compositional and if so, how many *primitives* there are (Wilks, 1977; Wierzbicka, 1996; Goddard, 1998).

[1] Roget's *thesaurus* (Roget, 1852), Miller and Fellbaum's *WordNet* (Fellbaum, 1998) and Longman's *Language Activator* (Summers, 1993), being notable exceptions (For more details, see next section).

[2] Of course, this does not preclude, that we may have to use *words* to refer to them in a concept-based query.

[3] While we agree with Polguère theoretically when he pleads for dictionary neutrality with regard to lexical access (Polguère, 2006), from a practical point of view the situation is obviously quite different for the speaker and listener, even if both of them draw on the same resource.

Coling 2008: Proceedings of the workshop on Cognitive Aspects of the Lexicon (COGALEX 2008), pages 9–17
Manchester, August 2008

Neither do we know how to represent them. Yet, there are ways around this problem as we will show. Whether *concepts* and *words* are organized and accessed differently is a question we cannot answer here. We can agree though on the fact that getting information concerning words is fairly unproblematic when reading, at least in the case of most western languages. Words can generally be found easily in a dictionary, provided the user knows the spelling, the alphabet and how to build lemma starting from an inflected form. Unlike *words*, which are organized alphabetically (in western languages) or by form (stroke counts in Chinese), *concepts* are organized topically: they are clustered into functional groups according to their role in real world, or our perception of it.

Psychologist have studied the difficulties people have when trying to produce or access words (Aitchinson, 2003). In particular, they have studied the *tip-of-the-tongue phenomenon* (Brown and McNeill, 1996) and the effects an input can have on the *quality* of an output (error analysis (Cutler, 1982)) and on the *ease* of its production: positive or negative priming effect (activation/inhibition). Obviously, these findings allow certain conclusions, and they might guide us when developing tools to help people find the needed word. In particular, they reveal two facts highly relevant for our goal:

1. even if people fail to access a given word, they might know a lot about it: *origin*, *meaning* (word definition, role played in a given situation), *part of speech*, *number of syllables*, *similar sounding* words, etc. Yet, despite all this knowledge, they seem to lack some crucial information to be able to produce the phonetic form. The word gets blocked at the very last moment, even though it has reached the *tip-of-the-tongue*. This kind of nuisance is all the more likely as the target word is rare and primed by a similar sounding word.

2. unlike words in printed or electronic dictionaries, words in our mind may be inexistent as tokens. What we seem to have in our minds are decomposed, abstract entities which need to be synthesized over time.[4] According to Levelt (Levelt, 1996) the generation of words (synthesis) involves the following stages: conceptual preparation, lexical selection, phonological- and phonetic encoding, articulation. Bear in mind that having performed 'lexical selection' does not imply access to the phonetic form (see the experiments on the *tip-of-the-tongue phenomenon*).

What can be concluded from these observations? It seems that underspecified input is sufficiently frequent to be considered as normal. Hence we should accept it, and make the best out of it by using whatever information is available (accessible), no matter how incomplete, since it may still contribute to find the wanted information, be it by reducing the search space. Obviously, the more information we have the better, as this reduces the number of words among which to choose.

2 Related work and goal

While more dictionaries have been built for the reader than for the writer, there have been some onomasiological attempts as early as in the middle of the 19th century. For example, Roget's *Thesaurus* (Roget, 1852), T'ong's *Chinese and English instructor* (T'ong, 1862), or Boissiere's *analogical dictionary* (Boissière, 1862).[5] Newer work includes Mel'čuk's *ECD* (Mel'čuk et al., 1999), Miller and Fellbaum's *WordNet* (Fellbaum, 1998), Richardson and Dolan's *MindNet* (Richardson et al., 1998), Dong's *HowNet* (Dong and Dong, 2006) and Longman's *Language Activator* (Summers, 1993). There is also the work of

[4] This may be very surprising, yet, this need not be the case if we consider the fact that speech errors are nearly always due to competing elements from the same level or an adjacent one, unless they are the result of a surrounding concept which has been activated, or which is about to be translated into words. Put differently, we do not store words at all in our mind, at least not in the layman's or lexicographer's sense who consider word-forms and their meanings as one. If we are right, than rather continue to consider the human mind as a *word store* we could consider it as a *word factory*. Indeed, by looking at some of the work done by psychologists who try to emulate the mental lexicon (for a good survey see (Harley, 2004), pages 359-374) one gets the impression that words are synthesized rather than located and read out. Taking a look at all this work, generally connectionist models, one may conclude that, rather than having words in our mind we have a set of more or less abstract features (concepts, syntactic information, phonemes), distributed across various layers, which need to be synthesized over time. To do so we proceed from abstract meanings to concrete sounds, which at some point were also just abstract features. By propagating energy rather than data (as there is no message passing, transformation or cumulation of information, there is only activation spreading, that is, changes of energy levels, call it weights, electronic impulses, or whatever), that we propagate signals, activating ultimately certain peripheral organs (larynx, tongue, mouth, lips, hands) in such a way as to produce movements or sounds, that, not knowing better, we call words.

[5] For a more recent proposal see (Robert et al., 1993).

(Fontenelle, 1997; Sierra, 2000; Moerdijk, 2008), various *collocation dictionaries* (BBI, OECD) and Bernstein's *Reverse Dictionary*.[6] Finally, there is M. Rundell's MEDAL, a thesaurus produced with the help of Kilgariff's Sketch Engine (Kilgarriff et al., 2004).

As one can see, a lot of progress has been accomplished over the last few years, yet more can be done, especially with regard to unifying *linguistic* and *encyclopedic* knowledge. Let's take an example to illustrate our point.

Suppose, you were looking for a word expressing the following ideas: 'superior dark coffee made from beans from Arabia', and that you knew that the target word was neither *espresso* nor *cappuccino*. While none of this would lead you directly to the intended word, *mocha*, the information at hand, i.e. the word's definition or some of its elements, could certainly be used. In addition, people draw on knowledge concerning the *role* a concept (or word) plays in language and in real world, i.e. the associations it evokes. For example, they may know that they are looking for a *noun* standing for a *beverage* that *people* take under certain circumstances, that the *liquid* has certain properties, etc. In sum, people have in their mind an encyclopedia: all words, concepts or ideas being highly connected. Hence, any one of them has the potential to evoke the others. The likelihood for this to happen depends, of course, on factors such as *frequency* (associative strength), *distance* (direct vs. indirect access), *prominence* (saliency), etc.

How is this supposed to work for a dictionary user? Suppose you were looking for the word *mocha* (target word: t_w), yet the only token coming to your mind were *computer* (source word: s_w). Taking this latter as starting point, the system would show all the connected words, for example, *Java, Perl, Prolog* (programing languages), *mouse, printer* (hardware), *Mac, PC* (type of machines), etc. querying the user to decide on the direction of search by choosing one of these words. After all, s/he knows best which of them comes closest to the t_w. Having started from the s_w 'computer', and knowing that the t_w is neither some *kind of software* nor a *type of computer*, s/he would probably choose *Java*, which is not only a *programming language* but also an *island*. Taking this latter as the new starting point s/he might choose *coffee* (since s/he is looking for some kind of *beverage*, possibly made from an ingredient produced in Java, coffee), and finally *mocha*, a type of *beverage* made from these beans. Of course, the word *Java* might just as well trigger *Kawa* which not only rhymes with the s_w, but also evokes *Kawa Igen*, a javanese volcano, or familiar word of *coffee* in French.

As one can see, this approach allows word access via multiple routes: there are many ways leading to Rome. Also, while the distance covered in our example is quite unusual, it is possible to reach the goal quickly. It took us actually very few moves, four, to find an indirect link, between two, fairly remotely related terms: *computer* and *mocha*. Of course, *cyber-coffee* fans might be even quicker in reaching their goal.

3 The lexical matrix revisited

The main question that we are interested in here is how, or in what terms, to index the dictionary in order to allow for quick and intuitive access to words. Access should be possible on the basis of meaning (or meaning elements), various kinds of associations (most prominently 'syntagmatic' ones) and, more generally speaking, underspecified input. To this end we have started to build an *association matrix* (henceforth AM), akin to, yet different from G. Miller's initial proposal of WN (Miller et al., 1990). He suggested to build a lexical matrix by putting on one axis all the *forms*, i.e. words of the language, and on the other, their corresponding *meanings*. The latter being defined in terms of synsets. The corresponding meaning-form relations are signaled via a boolean (presence/absence). Hence, looking at the intersection of meanings and forms, one can see which meanings are expressed by, or converge toward what forms, or conversely, what form expresses which meanings. Whether this is the way WN is actually implemented is not clear to us, though we believe that it is not. Anyhow, our approach is different, and we hope the reader will understand in a moment the reasons why.

We will also put on one axis all the form elements, i.e. the *lemmata* or expressions of a given language (we refer to them as *target words*, henceforth t_w). On the other axis we will place the *triggers* or *access-words* (henceforth a_w), that is, the words or concepts capable and likely to evoke the t_w. These are typically the kind of words psy-

[6] There is also at least one electronic incarnation of a dictionary with reverse access, combining a dictionary (WordNet) and an encyclopedia (Wikipedia) (http://www.onelook.com/reverse-dictionary.shtml).

chologists have gathered in their association experiments (Jung and Riklin, 1906; Deese, 1965; Schvaneveldt, 1989). Note, that instead of putting a boolean value at the intersection of the t_w and the a_w, we will put *weights* and the *type of link* holding between the co-occurring terms. This gives us quadruplets. For example, an utterance like "this is the key of the door" might yield the a_w (key), the t_w(door), the link type l_t(part of), and a weight (let's say 15).

The fact that we have these two kinds of information is very important later on, as it allows the search engine to cluster by type the possible answers to be given in response to a user query (word(s) provided as input) and to rank them. Since the number of hits, i.e. words from which the user must choose, may be substantial (depending on the degree of specification of the input), it is important to group and rank them to ease navigation, allowing the user to find directly and quickly the desired word, or at least the word with which to continue search.

Obviously, different word senses (homographs), require different entries (bank-money vs bank-river), but so will synonyms, as every word-form, synonym or not, is likely to be evoked by a different key- or access-word (similarity of sound).[7]

Also, we will need a new line for every different relation between a a_w and a t_w. Whether more than one line is needed in the case of identical links being expressed by different linguistic resources (the lock *of* the door vs. the door*'s* lock vs. the door *has* a lock) remains an open empirical question.

Let us see quickly how our AM is supposed to work. Imagine you wanted to find the word for the following concept: *hat of a bishop*. In such a case, any of the following concepts or words might come to your mind: church, Vatican, abbot, monk, monastery, ceremony, ribbon, and of course rhyming words like: brighter, fighter, lighter, righter, tighter, writer,[8] as, indeed, any of them could remind us of the t_w: *mitre*. Hence, all of them are possible a_w.

Once this resource is built, access is quite straightforward. The user gives as input all the words coming to his mind when thinking of a given idea or concept,[9] and the system will display all connected words. If the user can find the item he is looking for in this list, search stops, otherwise it will continue, the user giving other words of the list, or words evoked by them.

Of course, remains the question of how to build this resource, in particular, how to populate the axis devoted to the trigger words, i.e. *access-keys*. At present we consider three approaches: one, where we use the words occurring in word definitions (see also, (Dutoit and Nugues, 2002; Bilac et al., 2004)), the other is to mine a well-balanced corpus, to find co-occurrences within a given window (Ferret and Zock, 2006), the size depending a bit on the text type (encyclopedia) or type of corpus. Still another solution would be to draw on the association lists produced by psychologists, see for example http://www.usf.edu/, or http://www.eat.rl.ac.uk.

Of course, the idea of using matrices in linguistics is not new. There are at least two authors who have proposed its use: M. Gross (Gross, 1984) used it for coding the syntactic behavior of lexical items, hence the term *lexicon-grammar*, and G. Miller, the father of WN (Miller et al., 1990) suggested it to support lexical access. While the former work is not relevant for us here, Miller's proposal is. What are the differences between his proposal and ours? There are basically four main differences:

1. we use, collocations or *access-words*, i.e a_{ws} rather than *synsets*; Hence, any of the following a_{ws} (cat, grey, computer device, cheese, Speedy Gonzales) could point toward the t_w 'mouse', none of them are part of the meaning, leave alone synonyms.

2. we mark explicitly the *weight* and the *type of link* between the t_w and the a_w (isa, part_of, etc.),[10] whereas WN uses only a binary value. Both the *weight* and *link* are necessary information for ranking and grouping, i.e. navigation.

3. our AM is corpus-sensitive (see below), hence, we can, at least in principle, accommo-

[7] Take, for example, the nouns *rubbish* and *garbage* which can be considered as synonyms. Yet, while the former may remind you of a *rabbit* or (horse)-*radish*, the latter may evoke the word *cabbage*.

[8] The question, whether rhyming words should be computed is not crucial at this stage.

[9] The quantifier *all* shouldn't be taken too literally. What we have in mind are "salient" words available in the speaker's mind at a given moment

[10] Hence, if several links are possible between the t_w and the a_w, several cells will be used. Think of the many possible relations between a city and a country, example: *Paris* and *France* (part of, biggest city of, located in, etc.)

date the fact that a speaker is changing topics, adapting the weight of a given word or find a more adequate a_w in this new context. Think of 'piano' in the contexts of a concert or moving your household. Only the latter would evoke the notion of weight.

4. relying on a corpus, we can take advantage of *syntagmatic associations* (often encyclopedic knowledge), something which is difficult to obtain for WN.

4 Keep the set of lexical candidates small

Here and in the next section we describe how the idea of the AM has been computationally dealt with. The goal is to reduce the number of hits, i.e. possible t_{ws} (output), as a function of the input, i.e, the number of relevant a_{ws} given by the speaker/writer. To achieve this goal we apply lexical functions to the a_{ws}, considering the intersection of the obtained sets to be the relevant t_{ws}.

4.1 Lexical Functions

The usefulness of *lexical functions* for linguistics in general and for language production in particular has been shown by Mel'čuk (Mel'čuk, 1996). We will use them here, as they seem to fit also our needs of information extraction or lexical access.

Mel'čuk has coined the term *lexical functions* to refer to the fact that two terms are systematically related. For example, the lexical function *Gener* refers to the fact that some term (let's say ‹*cat*›) can be replaced by a more general term (let's say ‹*animal*›).

Lexical functions encode the combinability of words. While 'big' and 'strong' express the same idea (intensity, magnitude), they cannot be combined freely with any noun: *strong* can be associated with *fever*, whereas *big* cannot. Of course, this kind of combinability between lexical terms is language specific, because unlike in English, in French one can say *grosse fièvre* or *forte fièvre*, both being correct (Schwab and Lafourcade, 2007). Our AM handles, of course these kind of functions. Here is a list of some of them:

- *paradigmatic associations*: hypernymy (‹*cat*› - ‹*animal*›), hyponymy, synonymy, or antonymy,...;

- *syntagmatic associations*: collocations (‹*fear*› being associated with ‹*strong*› or ‹*little*›);

- *morphological relations* ie. terms being derived from another part of speech: applying the *change-part-of-speech* lexical function f_{cpos} to ‹*garden*› will yield: $f_{cpos}(\text{‹}garden\text{›}) = \{\text{‹}to\ garden\text{›}, \text{‹}gardener\text{›}, \dots\}$

- *sound-related items*: homophones, rhymes.

4.2 Assumptions concerning search

The purpose of using lexical functions is to reduce the number of possible outcomes from which the user must choose. The list contains either the t_w or another promising a_w the user may want use to continue search. Hence, lexical functions are useful for search provided that:

1. the speaker/writer is able to specify the kind of relations s/he wants to use. The problem here lies in the nature and number of the functions, some of them being very well specified, while others are not.

2. the larger the number of trigger words the smaller the list of words from which to choose: the speaker/writer can add or delete words to broaden or narrow the scope of his/her query.

These hypotheses are being modeled by using set properties of lexical functions. The idea is to apply all functions, or a selection of them, to the a_{ws} and to give the speaker/writer the intersection as result (see section 5.3.5 for an example)

5 Experiment

We have started with a simple, preliminary experiment. Only one lexical function was used: neighborhood (henceforth f_{neig}). Let f_{neig} be the function producing the set of co-occurring terms within a given window (sentence or a paragraph).[11] The result produced by the system and returned to the user is the intersection of the application of f_{neig} to the a_{ws}. In the next section we explain how this function is applied to two corpora (Wordnet and Wikipedia), to show their respective qualities and shortcomings for this specific task.

5.1 WordNet

5.1.1 Description

WordNet (henceforth WN) is a lexical database for English developed under the guidance of G.

[11] The scope or window size will vary with the text type (normal text vs. encyclopedia). The optimal size is at this point still an empirical question.

Miller (Miller et al., 1990). One of his goals was to support lexical access akin to the human mind, association-based. *Knowledge* is stored in a network composed of nodes and links (nodes being words or concepts and the links are the means of connecting them) and *access to knowledge*, i.e. search, takes place by entering the network at some point and follow the links until one has reached the goal (unless one has given up before). This kind of *navigation* in a huge conceptual/lexical network can be considered equivalent to *spreading activation* taking place in our brain.

Of course, such a network has to be built, and navigational support must be provided to find the location where knowledge or words are stored. This is what Miller and his coworkers did by building WN. The resource has been built manually, and it contains at present about 150.000 entries.

The structure of the dictionary is different from conventional, alphabetical resources. Words are organized in WN in two ways. Semantically similar words, i.e. synonyms, are grouped as clusters. These sets of synonyms, called *synsets*, are then linked in various ways, depending on the kind of relationship they entertain with the adjacent synset. For example, their neighbors can be more *general* or *specific* (hyperonymy vs. hyponymy), they can be *part of* some reference object (meronymy: car-motor), they can be the *opposite* (antonymy: hot-cold), etc. While WN is a resource it can also be seen as a corpus.

5.1.2 Using WN as a corpus

There are many good reasons to use WN for learning f_n. For one, there are many extensions, and second, the one we are using, eXtended WN (Mihalcea and Moldovan, 2001) spares us the trouble of having to address issues like: (a) segmentation: we do not need to identify sentence boundaries ; (b) semantic ambiguity: words being tagged, we get good precision; (c) lemmatization: since only verbs, nouns, adjectives and adverbs are tagged, we need neither a stoplist nor a lemmatizer.

Despite all these qualities, two important problems remain nevertheless for this kind of corpus: (a) size: though, all words are tagged, the corpus remains small as it contains only 63.941 different words; (b) in consequence, the corpus lacks many *syntagmatic associations* encoding encyclopedic knowledge.

5.2 Using Wikipedia as corpus

Wikipedia is a free, multilingual encyclopedia, accessible on the Web.[12] For our experiment we have chosen the English version which of this day (12th of may 2008) contains 2,369,180 entries.

Wikipedia has exactly the opposite properties of WN. While it covers well encyclopedic relations, it is only raw text. Hence problems like text segmentation, lemmatisation and stoplist definition need to be addressed.

Our experiments with Wikipedia were very rudimentary, given that we considered only 1000 documents. These latter were obtained in response to the term ‹*wine*›, by following the links obtained for about 72.000 words.

5.3 Prototype

5.3.1 Building the resource and using it.

Building the resource requires processing a corpus and building the database. Given a corpus we apply our neighborhood function to a predetermined window (a paragraph in the case of encyclopedias).[13] The result, i.e. the co-occurrences, will be stored in the database, together with their *weight*, i.e. number of times two terms appear together, and the *type of link*. As mentionned above, both kinds of information are needed later on for *ranking* and *navigation*.[14]

At present, cooccurences are stored as triplets (t_w, a_w, *times*), where *times* represents the number of times the two terms cooccur in the corpus, the scope of coccurence being here the paragraph.

5.3.2 Processing of the Wikipedia page

For each Wikipedia page, a preprocessor converts HTML pages into plain text. Next, a part-of-speech tagger (http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/) is used to annotate all the words of the paragraph under consideration. This allows the filtering of all irrelevant words, to keep but a bag of words, that is, the nouns, adjectives, verbs and adverbs occuring in the paragraph. These words will be used to fill the triplets of our database.

[12] `http://www.wikipedia.org`

[13] The optimal window-size depends probably on the text type (encyclopedia vs. unformatted text). Yet, in the absence of clear criteria, we consider the optimal window-size as an open, empirical question.

[14] This latter aspect is not implemented yet, but will be added in the future, as it is a necessary component for easy navigation (Zock and Bilac, 2004; Zock, 2006; Zock, 2007).

5.3.3 Corpus Building

We start arbitrarily from some page (for our experiment, we have chosen "wine" as input), apply the algorithm outlined here above and pick then randomly a noun within this page to fetch with this input a new page on Wikipedia. This process is repeated until a given sample size is obtained (in our case 1000 pages). Of course, instead of picking randomly a noun, we could have decided to process all the nouns of a given page, and to add then incrementally the nouns of the next pages. Yet, doing this would have led us to privilege a specific topic (in our case 'wine') instead of a more general one.

5.3.4 Usage

We have developed a website in Java as a servlet. Interactions with humans are simple: people can add or delete a word from the current list (see *Input* in the figure on top of the next page). The example presented shows that with very few words, hence very quickly, we can obtain the desired word.

Given some input, the system provides the user with a list of words cooccuring with the a_{ws}. The output is an ordered list of words, the order depending on the overall score, i.e. number of cooccurrences between the a_w and the t_w. For example, if the a_{ws} 'wine' and 'harvest' co-occur with the t_w 'bunch' respectively 5 and 8 times, then the overall score of cooccurence of 'bunch' is 13: ((wine, harvest), bunch, 13). Hence, all words with a higher score will precede it, while those with a lower score will follow it.

5.3.5 Examples and Comparison of the results of the two corpora

Here below are the examples extracted from the WN corpus (see figure-1). Our goal was to find the word ‹*vintage*›. Trigger words are ‹*wine*› and ‹*harvest*›, yielding respectively 488 and 30 hits, i.e. words. As one can see ‹*harvest*› is a better access term than ‹*wine*›. Combining the two will reduce the list to 6 items. Please note that the t_w ‹*vintage*› is not among them, eventhough it exists in WordNet, which illustrates nicely the fact that storage does not guarantee accessibility (Sinopalnikova and Smrz, 2006).

Looking at figure-1 you will see that the results have improved considerably with Wikipedia. The same input, ‹*wine*› evokes many more words (1845 as opposed to 488). For ‹*harvest*› we get 983 hits instead of 30 (the intersection containing 62 words). Combining the two reduces the set to 45 items among which we will find, of course, the target word.

Input	WordNet		Wikipedia	
	488 words		**1845 words**	
	grape	sweet	alcoholic	country
	serve	france	god	characteristics
wine	small	fruit	regulation	grape
	dry	bottle	appellation	system
	produce	red	bottled	like
	bread	hold	christian	track
	…	…	…	…
	30 words		**983 words**	
	month	fish	produce	grain
	grape	revolutionary	autumn	farms
	calendar	festival	energy	cut
harvest	butterfish	dollar	combine	ground
	person	make	balance	rain
	wine	first	amount	rich
	…	…	…	…
	6 words		**45 words**	
	make	grape	grape	vintage
wine	fish	someone	bottle	produce
+harvest	commemorate	person	fermentation	juice
	…	…	Beaujolais	taste
			viticulture	France
			Bordeaux	vineyard
			…	…

Figure 1: Comparing two corpora (*eXtended WordNet* and *Wikipedia*) with various inputs

We hope that this example is clear enough to convince the reader that it makes sense to use real text as corpus to extract from it the kind of information (associations) people are likely to give when looking for a word.

6 Conclusion and perspectives

We have addressed in this paper the problem of word finding for speakers or writers. Concluding that most dictionaries are not well suited to allow for this kind of reverse access based on meanings (or meaning related elements, associations), we looked at work done by psychologists to get some inspiration. Next we tried to clarify which of these findings could help us build the dictionary of tomorrow, that is, a tool integrating linguistic and encyclopedic knowledge, allowing navigation by taking either or as starting point. While linguistic knowledge is more prominent for analysis (reading), encyclopedic facts are more relevant for production. We've presented then our ideas of how to build a resource, allowing lexical access based

on underspecified, i.e. imperfect input. To achieve this goal we've started building an AM composed of form elements (the words and expressions of a given language) and a_{ws}. The role of the latter being to lead to or to evoke the t_w. In the last part we've described briefly the results obtained by comparing two resources (WN and Wikipedia) and various inputs. Given the fact that the project is still quite young, only preliminary results can be shown at this point.

Our next steps will be to take a closer look at the following work: clustering of similar words (Lin, 1998), topic signatures (Lin and Hovy, 2000) and Kilgariff's sketch engine (Kilgarriff et al., 2004). We plan also to add other lexical functions to enrich our database with a_{ws}. We plan to experiment with corpora, trying to find out which ones are best for our purpose[15] and we will certainly experiment with the window size [16] to see which size is best for which text type. Finally, we plan to insert in our AM the relations holding between the a_w and the t_w. As these links are contained in our corpus, we should be able to identify and type them. The question is, to what extent this can be done automatically.

Obviously, the success of our resource will depend on the quality of the corpus, the quality of the a_{ws}, weights and links, and the representativity of all this for a given population. While we do believe in the justification of our intuitions, more work is needed to reveal the true potential of the approach. The ultimate judge being, of course, the future user.

[15]For example, we could consider a resource like ConceptNet of the Open Mind Common-Sense project (Liuh and Singh, 2004).

[16]For example, it would have been interesting to consider coocurrences beyond the scope of the paragraph, by considering the logical structure of the Wikipedia document. Anyhow, our experiment needs to be redone with more data than just 1000 pages, the size chosen here for lack of time. Indeed one could consider using the entire corpus of Wikipedia or mixed corpora

References

Aitchinson, Jean. 2003. *Words in the Mind: an Introduction to the Mental Lexicon (3d edition)*. Blackwell, Oxford.

Bilac, S., W. Watanabe, T. Hashimoto, T. Tokunaga, and H. Tanaka. 2004. Dictionary search based on the target word description. In *Proc. of the Tenth Annual Meeting of The Association for NLP (NLP2004)*, pages 556–559, Tokyo, Japan.

Boissière, P. 1862. *Dictionnaire analogique de la langue française : répertoire complet des mots par les idées et des idées par les mots*. Larousse et A. Boyer, Paris.

Brown, R. and D. McNeill. 1996. The tip of the tounge phenomenon. *Journal of Verbal Learning and Verbal Behaviour*, 5:325–337.

Cutler, A, editor, 1982. *Slips of the Tongue and Language Production*. Mouton, Amsterdam.

Deese, James. 1965. *The structure of associations in language and thought*. Johns Hopkins Press.

Dong, Zhendong and Qiang Dong. 2006. *HOWNET and the computation of meaning*. World Scientific, London.

Dutoit, Dominique and P. Nugues. 2002. A lexical network and an algorithm to find words from definitions. In van Harmelen, F., editor, *ECAI2002, Proc. of the 15th European Conference on Artificial Intelligence*, pages 450–454, Lyon. IOS Press, Amsterdam.

Fellbaum, Christiane, editor, 1998. *WordNet: An Electronic Lexical Database and some of its Applications*. MIT Press.

Ferret, Olivier and Michael Zock. 2006. Enhancing electronic dictionaries with an index based on associations. In *ACL '06: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, pages 281–288.

Fontenelle, Thierry. 1997. Using a bilingual dictionary to create semantic networks. *International Journal of Lexicography*, 10(4):275–303.

Goddard, Cliff. 1998. Bad arguments against semantic primitives. *Theoretical Linguistics*, 24(2-3):129–156.

Gross, Maurice. 1984. Lexicon-grammar and the analysis of french. In *Proc. of the 11th COLING*, pages 275–282, Stanford, CA.

Harley, Trevor. 2004. *The psychology of language*. Psychology Press, Taylor and Francis, Hove and New York.

Jung, Carl and F. Riklin. 1906. Experimentelle Untersuchungen über Assoziationen Gesunder. In Jung, C. G., editor, *Diagnostische Assoziationsstudien*, pages 7–145. Barth, Leipzig, Germany.

Kilgarriff, Adam, Pavel Rychlý, Pavel Smrž, and David Tugwell. 2004. The Sketch Engine. In *Proceedings of the Eleventh EURALEX International Congress*, pages 105–116, Lorient, France.

Levelt, Willem. 1996. A theory of lexical access in speech production. In *Proc. of the 16th Conference on Computational Linguistics*, Copenhagen, Denmark.

Lin, Chin-Yew and Eduard H. Hovy. 2000. The automated acquisition of topic signatures for text summarization. In *COLING*, pages 495–501. Morgan Kaufmann.

Lin, Dekang. 1998. Automatic retrieval and clustering of similar words. In *COLING-ACL*, pages 768–774, Montreal.

Liuh, H. and P. Singh. 2004. ConceptNet: a practical commonsense reasoning toolkit. *BT Technology Journal*.

Mel'čuk, I., N. Arbatchewsky-Jumarie, L. Iordanskaja, S. Mantha, and A. Polguère. 1999. *Dictionnaire explicatif et combinatoire du français contemporain Recherches lexico-sémantiques IV*. Les Presses de l'Université de Montréal, Montréal.

Mel'čuk, Igor. 1996. Lexical functions: A tool for the description of lexical relations in the lexicon. In Wanner, L., editor, *Lexical Functions in Lexicography and Natural Language Processing*, pages 37–102. Benjamins, Amsterdam/Philadelphia.

Mihalcea, Rada and Dan Moldovan. 2001. Extended WordNet: progress report. In *NAACL 2001 - Workshop on WordNet and Other Lexical Resources*, Pittsburgh, USA.

Miller, G. A., R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. 1990. Introduction to WordNet: An online lexical database. *International Journal of Lexicography, 3(4)*, pages 235–244.

Moerdijk, Fons. 2008. Frames and semagrams; Meaning description in the general dutch dictionary. In *Proceedings of the Thirteenth Euralex International Congress, EURALEX*, Barcelona.

Polguère, Alain. 2006. Structural properties of lexical systems: Monolingual and multilingual perspectives. Sidney. Coling workshop 'Multilingual Language Resources and Interoperability'.

Richardson, S., W. Dolan, and L. Vanderwende. 1998. Mindnet: Acquiring and structuring semantic information from text. In *ACL-COLING'98*, pages 1098–1102.

Robert, Paul, Alain Rey, and J. Rey-Debove. 1993. *Dictionnaire alphabetique et analogique de la Langue Française*. Le Robert, Paris.

Roget, P. 1852. *Thesaurus of English Words and Phrases*. Longman, London.

Schvaneveldt, R., editor, 1989. *Pathfinder Associative Networks: studies in knowledge organization*. Ablex, Norwood, New Jersey, US.

Schwab, Didier and Mathieu Lafourcade. 2007. Modelling, detection and exploitation of lexical functions for analysis. *ECTI Transactions Journal on Computer and Information Technology*, 2(2):97–108.

Sierra, Gerardo. 2000. The onomasiological dictionary: a gap in lexicography. In *Proceedings of the Ninth Euralex International Congress*, pages 223–235, IMS, Universität Stuttgart.

Sinopalnikova, Anna and Pavel Smrz. 2006. Knowing a word vs. accessing a word: Wordnet and word association norms as interfaces to electronic dictionaries. In *Proceedings of the Third International WordNet Conference*, pages 265–272, Korea.

Summers, Della. 1993. *Language Activator: the world's first production dictionary*. Longman, London.

T'ong, Ting-Kü. 1862. *Ying ü tsap ts'ün (The Chinese and English Instructor)*. Canton.

Wierzbicka, Anna. 1996. *Semantics: Primes and Universals*. Oxford University Press, Oxford.

Wilks, Yorick. 1977. Good and bad arguments about semantic primitives. *Communication and Cognition*, 10(3–4):181–221.

Zock, Michael and Slaven Bilac. 2004. Word lookup on the basis of associations : from an idea to a roadmap. In *Workshop on 'Enhancing and using electronic dictionaries'*, pages 29–35, Geneva. COLING.

Zock, Michael. 2006. Navigational aids, a critical factor for the success of electronic dictionaries. In Rapp, Reinhard, P. Sedlmeier, and G. Zunker-Rapp, editors, *Perspectives on Cognition: A Festschrift for Manfred Wettler*, pages 397–414. Pabst Science Publishers, Lengerich.

Zock, Michael. 2007. If you care to find what you are looking for, make an index: the case of lexical access. *ECTI, Transaction on Computer and Information Technology*, 2(2):71–80.