Journal of Universal Computer Science, vol. 8, no. 12 (2002), 1047-1064
submitted: 12/4/02, accepted: 21/12/02, appeared: 28/12/02 © J.UCS

On the Semiautomatic Generation of WordNet Type Synsets and Clusters

Florentina Hristea
(University of Bucharest, Romania
fhristea@mailbox.ro)

Abstract: WordNet (WN) is a lexical knowledge base, first developed for English and then adopted for several Western European languages, which was created as a machine-readable dictionary based on psycholinguistic principles. Our paper is an attempt to discuss the semiautomatic generation of WNs for languages other than English, a topic of great interest since the existence of such WNs will create the appropriate infrastructure for advanced Information Technology systems. Extending the algorithmic approach proposed in [Nikolov and Petrova, 01] we introduce a semiautomatic method based on heuristics for generating noun and adjective synsets and clusters. This choice of involved parts of speech is determined by the fact that nouns and adjectives have completely different organizations in WN: the hierarchy and the N-dimensional hyperspace respectively. Our approach to WN generation relies on so-called "class methods", namely it uses as knowledge sources individual entries coming from bilingual dictionaries and WN synsets, but at the same time demonstrates the need to combine such methods with structural ones.

Keywords: WordNet, e-set, synset, synset_id, cluster

Category: J - Computer Applications

1 Introduction

WordNet (WN) is a proposal for a more effective combination of traditional lexicographic information and modern high-speed computation. It is an on-line lexical reference system whose design is inspired by current psycholinguistic theories of human lexical memory. English nouns, verbs, adjectives and adverbs are organized into synonym sets (*synsets*), each representing one underlying concept. Different relations link the synonym sets, WN being organized according to semantic relations which are indicated by pointers between synsets.

WN primarily represents *an interactive lexical data base* developed, during the last 15 years, at Princeton University by a group of researchers led by George Miller. At the same time, WN can be viewed as *a semantic dictionary* since words are located according to conceptual affinities with other words, unlike the case of classical dictionaries where words are ordered alphabetically. Although it resembles *a thesaurus*, WN is much more useful to Artificial Intelligence applications since it is enriched with an impressive set of relations among words and word meanings. WN distinguishes between semantic relations and lexical relations, but the emphasis is on semantic relations between meanings. Therefore, unlike standard dictionaries, WN organizes lexical information in terms of word meanings, rather than word forms. WN maps word forms in word senses using the syntactic category as a parameter. Thus, words belonging to the same syntactic category which can be used to express the same meaning are grouped into a single set, called *synset*. Therefore the "building block" of WN is *a synonym set (synset) of all words that express a given concept*. Polysemous words belong to more than one synset. For instance, corresponding to the English word *computer*, two different meanings are defined in WN. It therefore belongs to two distinct synsets, as follows:

{computer,data processor,electronic computer,information processing system}
and
{calculator, reckoner, figurer, estimator, computer}.

In its most used version (ver.1.6), WN contains 129,509 English words organized in 99,643 synsets, with the network using a number of 229,152 nodes. Words and concepts are linked through a total of 299, 711 semantic relations. However, all numbers are approximate since WN continues to grow, version 1.7 now being available as well. The most ambitious feature of WN is most probably the semantic attempt and, in this respect, WN resembles a thesaurus more than a dictionary. It equally represents *an on-line thesaurus* and *a semantic network*.

The rich set of semantic relations established among synsets is what makes this semantic network so powerful and useful for various types of applications. Examples of semantic relations existing in WN are **synonymy**, used in order to form synsets, **hypernymy** and **hyponymy**, corresponding to the *isa* relation and to *reverse isa* respectively, **meronymy**, corresponding to the *part of* relation, the **causal** relation referring to verbs and others. Using the *isa* relation nouns and verbs are structured in WN as *hierarchies*. Adjectives and adverbs are organized according to a different structure - *the cluster*. As its authors note [Miller et. al., 90], the advantage of imposing this syntactic categorization on WN "is that fundamental differences in the semantic organization of these syntactic categories can be clearly seen and systematically exploited." Nouns are organized in lexical memory as topical hierarchies, verbs are organized by a variety of entailment relations, and adjectives and adverbs are organized as N-dimensional hyperspaces. Additionally, the typical properties of a specific concept are stated as a *gloss* attached to each of the concepts. The gloss includes a definition, one or more supplementary explanations and one or more examples.

WN has been recognized as a valuable resource in the *human language technology* and *knowledge processing* communities. The human language research community has encouraged the development of WNs for languages other than English, at the same time concentrating on the possibility of automatically generating such huge lexical data bases. The main reason for this is the desire and the necessity to create a **uniform ontological infrastructure across languages**. This can be achieved since, while concepts are language dependent, the basic set of relations that link the concepts remains the same. This means that the inference algorithms for extracting information remain the same. The existence of such an uniform ontological infrastructure across languages will therefore simplify **machine translation** from a language to another and will facilitate the use of the same **reasoning schemes** and **algorithms** developed in conjunction with the American WN.

The present paper concentrates on the important and up-to-date topic of automatic generation of WNs for languages other than English. The approach to WN generation consists of a semiautomatic method based on heuristics which belongs to the so-called "class methods" [Atserias et. al., 97]. It therefore uses individual entries coming from bilingual dictionaries and WN synsets as knowledge sources, and hence the success of our methods depends directly on the availability of such comprehensive bilingual dictionaries and WN synsets.

The basic translation algorithm (Algorithm 2.1 of the present paper) will be using so-called "elementary sets", a concept introduced in [Nikolov and Petrova, 00]. Algorithm 2.1, which is described in [Nikolov and Petrova, 01], will be further completed by Algorithm 2.2, proposed in the present paper, and which performs a backtracking action (step 1) in order to obtain as final output the foreign synset corresponding to the given English one. It should be noted that the Bulgarian authors who first describe Algorithm 2.1 [Nikolov and Petrova, 01], having as output a sorted list of elementary sets, make no comment whatsoever as to how they obtain the final foreign synset, in their case the final Bulgarian noun synset. One can easily assume that it is manually obtained by linguists using the output of Algorithm 2.1. It is the concern of the present paper to automatize the process of creation of a foreign WN type synset to the largest extent that this is possible, and our comments concerning output obtained in the case of the Romanian language will be made within this type of framework.

When referring to the output of Algorithm 2.1 in the case of the Bulgarian language, the same authors [Nikolov and Petrova, 01] do not specify what evaluation function they have been taking into account. Several evaluation functions are described, but no recommendation whatsoever is made with respect to this issue. The present paper recommends the use of a specific evaluation function which is described in §2.

Finally, to the praise of the mentiond authors, who are only concerned with obtaining "a core of Bulgarian noun synsets", it turns out that their algorithm can be extended (more or less successfully) to the general case of any foreign language (not just Bulgarian). Additionally, it is our belief that Algorithm 2.1 can be successfully used in the case of all other (three) parts of speech that WN deals with, provided that it is modified accordingly. Such modifications should take into account the typical semantic relations implemented in WN with regard to each part of speech, thus combining the class method used in the case of nouns with a structural approach to WN generation (see the enrichment technique proposed in the case of adjectives in §3.2 of the present paper).

Since in WN adjectives have a completely different organization than nouns - the N-dimensional hyperspace - our study concerning this part of speech is further extended by taking into account the semiautomatic generation of foreign adjective clusters. At this point our approach again makes the necessary links between class methods and structural ones (namely those that take profit of the WN structure). Algorithm 3.1, proposed in §3.3 of the present paper, represents a first approach to semiautomatic generation of foreign adjective clusters which does not make use of monolingual resources but only of bilingual ones, namely bilingual dictionaries in electronic format.

2 The Translation Algorithm

The algorithm for translating a given English synset into the corresponding synset in a language other than English will be using so-called "elementary sets" or **e-sets**, a concept introduced in [Nikolov and Petrova, 00]. An e-set corresponds to a monosemous reading (sense) of a word and can be defined as follows:

Definition 2.1

An *e-set* relative to a word is the set of synonyms corresponding to a specific monosemous reading (sense) of that word.

Let us denote by EW any English word and by FW any foreign word, namely a word of a language other than English. Let **eword** of sequence (1) be an EW, while *fword1, fword2* and *fword3* of the same sequence are its corresponding translation equivalents (according to the appropriate bilingual dictionary):

$$\textbf{eword} \quad \textit{fword1};\ \textit{fword2},\ \textit{fword3} \qquad (1)$$

In order to distinguish among *fword1, fword2* and *fword3* two different separators are used in standard paper dictionaries. A semicolon separates different meanings of a given word. A comma separates synonyms which refer to one and the same meaning of the word. (In this case *fword2* and *fword3* are synonyms). This is the form of a bilingual dictionary which will be used by the programs implementing the proposed translation algorithm. In the above example the involved e-sets are

$$\{\textit{fword1}\} \text{ and } \{\textit{fword2, fword3}\}.$$

The computer programs which implement the translation algorithm will generate the list of all e-sets of FWs corresponding to the meaning of all EWs occurring in a given English synset. The foreign synset corresponding to the studied

English one is formed of one or more of the generated e-sets (which can be adjoined). The "candidates" for inclusion in the foreign synset are *labeled e-sets,* namely those e-sets which contain *labeled words.*

In order to label the FWs belonging to the generated e-sets, we have decided to first label the EWs belonging to the English synset. These EWs will be labeled with integer numbers ranging from 1 to n (where n is the size of the synset, namely the number of words it contains), in the order of their occurrence. After labeling the EWs of the original synset, the FWs of the generated e-sets are looked up in the corresponding bilingual dictionary. Each time an EW of the given synset represents the translation, according to the dictionary, of a FW, the corresponding FW receives the label of that EW. If any word of a foreign e-set can be translated into a word of the English synset using the bilingual dictionary, the whole foreign e-set is moved to the "list of candidates". As noted in [Nikolov and Petrova, 01], when completed, this list of candidates is the most important preliminary result. The appropriate foreign synset must be a compilation of some e-sets belonging to this list. Various *evaluating functions* which sort the extracted e-sets and outline the most adequate ones have been developed. In order to define such evaluating functions let us refer to the following concepts:

Definition 2.2

The *label of an e-set* represents the number of labels assigned to the words belonging to that e-set.

Definition 2.3

An e-set is *unlabeled* if it contains no labeled words.

Any word can have one or more labels assigned to it (as well as no label at all). The most common evaluating function which is proposed in the literature [Nikolov and Petrova, 01] takes as argument an e-set and has a value given by the very label of that e-set. A variant of this evaluating function is that which divides the number representing the label of the e-set to the size of the same e-set.

As far as we are concerned, we have taken into consideration the evaluation function which is defined in this paragraph. Each EW belonging to the given English synset will have a label (represented by an integer number from 1 to n, where n is the size of the synset) and the labeling of the FWs belonging to the e-sets is performed according to this label. The labels of the foreign words which differ from the label of the corresponding EW will be considered as representing two points, while the others represent just one point. The value of the evaluation function relative to a specific e-set is given by the total number of points corresponding to that e-set divided by its size.

Having defined all necessary concepts, one can now state the algorithm for generating the foreign *e-sets* corresponding to a given English synset:

Algorithm 2.1

Input: The file containing the English synsets and the two files representing the two bilingual dictionaries (for instance, the English-French and the French-English dictionary respectively).

1. Create (by consulting the appropriate bilingual dictionary) the e-sets corresponding to each word of the given English synset.

2. Label the English words belonging to the given English synset.

3. Label each of the e-sets generated in Step 1.

4. Remove all unlabeled e-sets.

5. Evaluate the e-sets (using the assigned labels and an evaluating function).

Output: The sorted list of e-sets corresponding to the given English synset.

The translations in the foreign language of the words occurring in the English synset are extracted from the bilingual dictionary as follows:

eword*1* meaning*11*; meaning*12*; $\cdots$; meaning$1m_1$
....
eword$n$ meaning$n1$; meaning$n2$; $\cdots$; meaningnm_n

The set of e-sets generated by Algorithm 2.1 is of the following form:

$$\{\{\text{meaning}ij\} \mid 1 <= i <= n,\ 1 <= j <= m_i\}.$$

The *foreign synset* will be generated using this set.

In the automatic generation of the *foreign synset* corresponding to a given English synset we shall also take into account

Remark 2.1

Of all possible meanings of a word, only one refers to a specific concept (to which a synset corresponds).

Using the sorted list of e-sets generated by Algorithm 2.1 (namely the evaluated e-sets), the meaning (elementary set) evaluated with the highest value will be chosen corresponding to each English word. Let this meaning, corresponding to *ewordj*, be *meaningj*i_j.

The *foreign synset* will be generated using the e-sets obtained by means of Algorithm 2.1, taking into account Remark 2.1 and according to

Algorithm 2.2

Input: The sorted list of e-sets generated by Algorithm 2.1 corresponding to the given English synset [*eword1, eword2,...,ewordn*].

1. Compute the foreign synset as having the following form:

$$\{meaning1i_1\} \cup \{meaning2i_2\} \cup \ldots \{meaningni_n\}, 1 \leq i_j \leq m_j, \qquad \forall\, j = \overline{1,n}.$$

2. Delete words occurring in more than one e-set from this union, such that each word will occur just once.

Output: The foreign synset corresponding to the given English synset.

It has now become obvious that our approach to WN generation belongs to the class of semiautomatic methods based on heuristics. As it is well known [Atserias et. al., 97] such heuristics can belong to two main categories: one in which the corresponding heuristics rely on information found in the bilingual dictionaries and the structure of WN, another containing heuristics that rely on the genus information extracted from the monolingual dictionary. Obviously, the heuristic which is used here belongs to the first mentioned category since our generation method does not use monolingual resources (with the exception of WN itself) but relies solely on bilingual dictionaries (in electronic format).

Algorithms 2.1 and 2.2 have been implemented in Prolog and tested by us, with very good results, in the case of *Romanian nouns.* In order to test the algorithms, we have used fragments of bilingual dictionaries in electronic format. When working with a semantic network like WN the richness of the bilingual dictionaries which are used is of the essence. Due to the imperfection of existing Romanian - English and English - Romanian dictionaries in electronic format (see, for instance, www.castingsnet.com/dictionaries), and in order to ensure the most possible accurate testing, we have generated our own fragments of electronic bilingual dictionaries, using some of the most complete existing paper ones [Leviţchi, 73], [Leviţchi et. al., 74]. The compiled Romanian-English and English-Romanian dictionaries used in our tests can be seen at

http://phobos.cs.unibuc.ro/roric/wn/r_e.dict
and
http://phobos.cs.unibuc.ro/roric/wn/e_r.dict

respectively. We have randomly chosen a number of 200 English noun synsets for which we have automatically generated the corresponding Romanian ones. Since most English synsets contain two words, our data sample was chosen according to the same pattern. Thus, out of the 200 considered English synsets, 179 contained two English nouns, 4 synsets contained 3 English nouns and 17 synsets contained more than 3 English nouns (between 4 and 7 words). The number of e-sets involved in the experiment was of 616. Several English synsets containing just one noun have been subsequently taken into consideration. All tests performed have been using the original WN 1.6 in its Prolog-readable format.

The generated Romanian synsets were validated by Romanian linguists using the latest bilingual dictionaries and the corresponding gloss indicated in the American WN. As noted before, this gloss contains the explanation corresponding to a synonym string, thus containing the meronym, or the "mother" concept from a higher level in the hierarchy. Actually, when testing the translation algorithm relatively to Romanian nouns, we have noticed that, in several cases, Algorithm 2.2 has generated more than one Romanian synset corresponding to the given English one. This was the case when Algorithm 2.1 had as output a list of e-sets (corresponding to different meanings of the same word) that had been evaluated with the same value. Each such e-set then represented a candidate and led to a different Romanian, or, in general, foreign synset. In such cases the correct foreign synset will be chosen from the list of synsets generated by Algorithm 2.2 according to the gloss of the given English synset. The computer program implementing Algorithm 2.2 must therefore provide as output the gloss as well, since it is necessary in the validation performed by linguists.

When performing tests for Romanian nouns it turned out that for 87% of the considered English synsets the generated Romanian ones were correct. In most other cases the algorithm has generated several Romanian synsets, among which the correct one could be found. In those cases when the English synsets did not have correct Romanian counterparts it was mostly because of wrong or missing data in the bilingual dictionaries. We consider this result a very successful one, since it is well known that one can not work 100% automatically when dealing with linguistic resources.

Also in order to facilitate the experiment, when choosing our sample of English synsets a necessary step was that of removing the synsets with proper names, compounds and collocations. These should be dealt with separately and with a more significant contribution on the part of the linguists. However, the presented algorithms are sufficient for building a *core* of synsets corresponding to all four parts of speech in more or less any language other than English, provided that good bilingual dictionaries in electronic format exist for the specific foreign language involved.

As it is noted in [Nikolov and Petrova, 01], the greatest advantage of Algorithm 2.1 is the ability to create synsets which may include foreign words that would not be extracted from the input resource at the first step of the work. Thus, even if a foreign word occurs in the English-Romanian dictionary, for instance, but is missing from the Romanian-English one, there is still a big chance for this word to be included in the final resulting synset. (The only necessary condition for this is the presence in the list of candidates of an e-set which includes that word). This is a very important fact considering how incomplete bilingual dictionaries usually are. This algorithm, therefore, does not represent a simple mirror translation.

Obviously, when using Algorithms 2.1 and 2.2 for specific languages, various difficulties will occur according to what is typical of each language at morphological and derivational level. In the case of the Romanian language, we have come to the conclusion that, in those, more interesting cases, in which the bilingual dictionaries are not to blame, the main difficulties which occur when automatically translating the English synsets into Romanian ones were generated by loan translation and by the fact that the polysemy of many English words is greatly superior to that of the corresponding Romanian words. The latter situation affects especially English synsets containing a single word. For instance, the English word "feature" having the meaning of "an article of merchandise that is displayed or advertised more than other articles" has no correspondent in Romanian. No single word with this meaning exists. We are therefore obliged to perform translation using a group of words (a gloss), while the English synset containing the sole word "feature" which refers to this concept will have no Romanian counterpart. In this case, the computer program did not work correctly. It is, once again, a situation which affects primarily English synsets containing a single word. This type of difficulty has suggested to us the enrichment technique which is proposed in §3.2 with regard to adjectives. Such enrichment, performed by means of the similarity relation, is not possible in the case of nouns since in WN this relation only holds for adjective synsets contained in adjective clusters.

In spite of such difficulties, however, we consider the presented translation algorithm as being appropriate for performing a semiautomatic extraction of the *core* of a foreign WN from the original WN 1.6, which we have been using, or from WN 1.7 (the latest version of WN). The most important issue here is the fact that Algorithms 2.1 and 2.2 do not depend on the type of part of speech involved in the translation. It is therefore natural to expect similar or even better results than the ones obtained for nouns when testing with regard to other parts of speech, such as the adjective. Especially since adjectives are much less polysemous than nouns.

In what follows, we shall establish how this general algorithm must be enriched in order for it to perform the semiautomatic generation of **adjective synsets** and **clusters** in languages other than English.

3 Generation of adjective synsets and clusters

3.1 Adjectives in WordNet

WN divides adjectives into two major classes : *descriptive* and *relational.* Chromatic *color adjectives* are regarded as a special case.

A *descriptive adjective* is one that ascribes a value of an attribute to a noun. That is to say, *x is Adj* presupposes that there is an attribute A such that $A\,(x) = Adj$. For instance, *low* and *high* are values for the attribute HEIGHT. WN contains pointers between descriptive adjectives and the noun synsets that refer to the appropriate attributes.

The *semantic organization* of descriptive adjectives is entirely different from that of nouns. The hyponymic relation that generates nominal hierarchies in the case of nouns is not available for adjectives. The semantic organization of adjectives is more naturally thought of as an abstract hyperspace of N dimensions rather than as a hierarchical tree. The basic semantic relation among descriptive adjectives is *antonymy.*

The importance of antonymy in the organization of descriptive adjectives is understandable when it is recognized that the function of these adjectives is to express values of attributes, and that nearly all attributes are bipolar. Antonymous adjectives express opposing values of an attribute. For example, the antonym of *heavy* is *light,* which expresses a value at the opposite pole of the WEIGHT attribute. In WN this binary opposition is represented by reciprocal labeled pointers: *heavy!*→*light* and *light!*→*heavy.* In the Prolog implementation of WN, which we have been using, the **ant** operator specifies antonymous words and all Prolog facts using this operator are included in the file **wn_ant.pl.** Descriptive adjectives that do not have direct antonyms are said to have indirect antonyms by virtue of their semantic similarity to adjectives that do have direct antonyms. A similarity pointer was used to indicate that the adjectives lacking antonyms are similar in meaning to adjectives that do have antonyms. In the Prolog implementation of WN the **sim** operator specifies that two synsets are similar in meaning and all Prolog facts using this operator are included in the file **wn_sim.pl.**

Descriptive adjectives therefore ascribe to their head nouns values of (typically) bipolar attributes and consequently are organized in terms of *binary oppositions* (*antonymy*) and *similarity of meaning (synonymy).*

Gross, Fischer, and Miller (1989) proposed that adjective synsets be regarded as *cluster of adjectives* associated by semantic similarity to a focal adjective that relates the cluster to a contrasting cluster at the opposite pole of the attribute. Also Gross, Fischer and Miller distinguish direct antonyms like *heavy/light*, which are conceptual opposites that are also lexical pairs, from indirect antonyms, like *heavy/weightless*, which are conceptual opposites that are not lexically paired. Under this formulation, all descriptive adjectives have antonyms; those lacking direct antonyms have indirect antonyms, i.e. are synonyms of adjectives that have direct antonyms.

In WN direct antonyms are represented by the antonymy pointer '!→'; indirect antonyms are inherited through similarity, which is indicated by the similarity pointer, '&→'. The configuration that results is illustrated in Figure 1 for the cluster of adjectives around the direct antonyms, *wet/dry*, which define the attribute WETNESS or MOISTNESS, an example often used by various authors. When analyzing this cluster of adjectives, one sees that *moist*, for instance, does not have a direct antonym, but its indirect antonym can be found via the path *moist*&→*wet!*→*dry.*

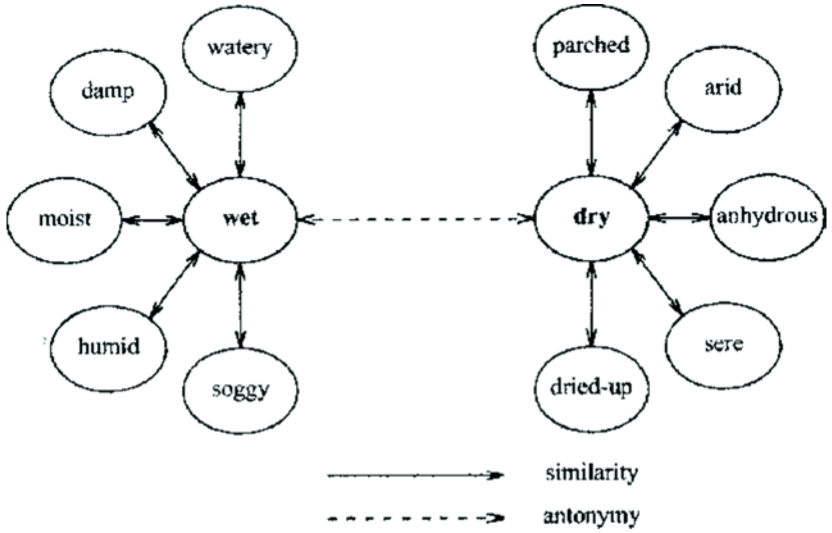

Figure 1. Bipolar Adjective Structure

Corresponding to Figure 1, which intuitively presents the structure of an adjective cluster in WN, one obtaines the following bipolar cluster having as head the antonym pair *wet/dry*:

[{ [WET, DRY, !] watery,& damp,& moist,& humid,& soggy,& }

{ watery, tearful, teary, wet, & }

{ damp, wet,& }

{ moist, wet,& }

{ humid, muggy, steamy, sticky, sultry, wet,& }

{ soggy, saturated, sodden, waterlogged, wet,& }

-

{ [DRY, WET, !] parched,& arid,& anhydrous,& sere,& dried-up,&}

{ parched, dehydrated, desiccated, dry,&}

{ arid, waterless, dry,&}
{ anhydrous, dry,& ((chem) with all water removed)}
{ sere, shriveled, withered, wizened, dry,& (used of vegetation)}
{ dried-up, dry,& ("a dry water hole") }].

One can notice that a cluster has two distinct parts. Each half of the cluster starts with a synset called *head synset.* The first two items of the head synset represent the antonym pair which defines the cluster and are capitalized. The antonym pair is followed by adjectives representing similarity pointers (watery,& damp,&, etc.), one to each synset similar in meaning of the corresponding half of the cluster. Each such synset contains a reciprocal pointer for returning to the head synset. One also notices that the similarity pointers occurring in the head synset are, in fact, words occurring on the first position within the synsets related by similarity with the synset to which the adjective having a direct antonym belongs. The two distinct cluster parts are separated by a hyphen and the entire structure is enclosed between square brackets. Each bipolar cluster stands alone, and coding is restricted to within-cluster relations.

The significance of existing exceptions is not obvious and we believe, together with the authors of WN [Miller et. al., 90], that the presented model accounts for the great majority of the English descriptive adjectives. The importance of the similarity relation is obvious.

In WN semantic relations are represented by a pair of *synset_id*s, in which the first *synset_id* is generally the source of the relation and the second is the target. A *synset_id* is a nine byte field in which the first byte defines the syntactic category of the synset and the remaining eight bytes are a *synset_offset*, indicating the byte offset in the file that corresponds to the syntactic category. In the Prolog version of the WN database, the *synset_id*s are used as unique synset identifiers. As it was already noted, in the Prolog implementation of WN the **sim** operator is used in order to designate the similarity relation, as in the example **sim**(302425348, 302425924). In general,

sim(*synset_id,synset_id).*

is a Prolog fact specifying that the second synset is similar in meaning to the first synset. This means that the second synset is a satellite of the first synset, which is the cluster head. This relation only holds for adjective synsets contained in adjective clusters.

Relational adjectives in WN are assumed to be stylistic variants of modifying nouns and so are cross-referenced to the noun files. Relational adjectives, which were first discussed at length by Levi (1978), mean something like "of, relating/pertaining to, or associated with" some noun, and therefore play a role similar to that of a modifying noun (as in *atomic bomb*). Relational adjectives differ from descriptive adjectives in that they do not relate to an attribute.

Therefore they do not refer to a property of their head nouns. Since relational adjectives do not have antonyms, they cannot be incorporated into the clusters that characterize descriptive adjectives. WN maintains a separate file of relational adjectives with pointers to the corresponding nouns. Each synset consists of one or more relational adjectives, followed by a pointer to the appropriate noun.

In what follows, we shall be concerned with the semiautomatic generation of *adjective clusters* in languages other than English, and will therefore refer solely to descriptive adjectives, which can be organized as this type of structure.

3.2 Semiautomatic generation of adjective synsets

In order to translate English adjective synsets into a foreign language Algorithms 2.1 and 2.2 have been used. When translating from English to any other language the *id* which is associated to a synset is not modified. This means that the similarity relation existing between two English synsets will be maintained after performing the translation and will occur among the foreign language adjective synsets as well.

A special problem is posed by synsets containing a single word. In this case it is impossible to tell which meaning of the word was involved in the creation of the specific synset if one has access to no additional information. The meaning can be guessed only from the gloss. However, in such cases we have used a strategy which consists in enriching the given synset with new adjectives that suggest the meaning of the one occurring in this synset. The new words are obtained using the similarity relation that typically exists in WN among adjective synsets. Thus, in order to enrich the given synset with new words, the adjectives occurring on the first position within synsets semantically linked to the original one via the similarity relation have been chosen. These words have been appended to the original synset, starting from the second position. This idea was inspired by the way in which adjective clusters are organized and structured. At this point one feels the necessity of combining the present class method with a structural one (namely one that takes profit of the WN structure).

The necessary list of e-sets in connection with the given English synset will be generated using Algorithm 2.1. When creating the foreign adjective synset representing the translation of the given English one, Algorithm 2.2 will combine all maximally evaluated e-sets corresponding to each of the words occurring in the English synset. In those cases when more than one e-set will be maximally evaluated corresponding to the same English word, Algorithm 2.2 will generate more than one foreign synset. The final decision concerning the correct translation is then again made according to the gloss.

In order to illustrate how Algorithms 2.1 and 2.2 work in the case of adjective synsets let us consider the English synset having the *id* 302428719 and containing the unique adjective *sticky*. We shall perform the translation to Romanian of this synset. Let us note that the chosen target language is not essential for the point that we are trying to make here. The presented results are the output of various Prolog programs.

Since the given English synset contains only one word, it will be enriched as mentioned, according to the similarity relation. After searching the database one comes to the conclusion that the only similarity relation (denoted by the **sim** operator) is

$$\mathrm{sim}(302428719, 302425348).$$

as well as its symmetrical relation. The synset having $id = 302425348$ contains the unique adjective *wet.* The given English synset is therefore enriched with this adjective. The evaluated e-sets obtained corresponding to the enriched synset, when using the evaluation function mentioned in §2 for Algorithm 2.1, are the following:

```
evset (302428719, sticky, 1.0, [lipicios, cleios, vascos]).
evset (302428719, sticky, 1.0, [umed, cetos]).
evset (302428719, wet, 1.0, [umed, jilav, ud]).
evset (302428719, wet, 0.6666666666666666, [ploios, umed, igrasios]).
```

Here *evset* is an operator designating evaluated sets. The first field represents the synset *id*, the second is the ASCII text of the word as entered by the lexicographer, the third gives the value of the evaluation function and the last denotes the foreign evaluated set.

In this case the computer program implementing Algorithm 2.2 has the following output:

```
English synset: [sticky]
Gloss: (moist as with undried prespiration and with clothing sticking to
the body; "felt sticky and chilly at the same time")
Romanian synset: [[lipicios,cleios,vascos,umed,jilav,ud], [umed,cetos,jilav,ud]]
```

One notices that the output consists of two possible Romanian synsets. However, only one of them corresponds to the meaning of *sticky* which refers to the underlying concept of the synset having $id = 302428719$. The correct foreign (in this case Romanian) synset can be easily chosen according to the corresponding gloss.

Such enrichment with additional words coming from synsets related via similarity with the original one is not always necessary. However, when performed, the chances of empty foreign adjective synsets being obtained (due to the generation uniquely of unlabeled e-sets) are considerably reduced. This operation might produce a slight shift in meaning with respect to the underlying concept of the original English synset. However, only similar concepts are denoted by the involved relation, typical for descriptive adjectives, a fact which determines us to recommend the described strategy. Both translation with and without enrichment can be performed, giving linguists the opportunity to compare and to choose among the proposed foreign synsets, when equally taking into consideration the corresponding gloss.

3.3 Semiautomatic generation of adjective clusters

The translation of English adjective clusters is completely ensured by the translation of the English adjective synsets and by that of the **ant** relation (denoting antonyms). Since the translation of adjective synsets has already been discussed in § 3.2, let us now refer to the translation of the **ant** relation. This becomes a very important issue when taking into account the fact that antonym dictionaries in electronic format do not exist for a great number of languages.

In the Prolog version of the WN database, which we have been using, semantic relations are represented by a pair of *synset_id*s, in which the first *synset_id* is generally the source of the relation and the second is the target, as is the case with the already mentioned **sim** operator. If two pairs *synset_id, w_num* are present, the operator represents a lexical relation between word forms, where *w_num* specifies the word number for a specific word in a specific synset. If present, *w_num* indicates which word in the synset is being referred to. The **ant** operator, for instance, specifies antonymous words in the following form:

ant(*synset_id,w_num,synset_id,w_num).*

Thus, the significance of the following Prolog fact

ant(302425348, 1, 302429323, 1).

is that the first word of the synset having the *id* 302425348 and the first word of the synset having the *id* 302429323 are *direct antonyms.*
This is a lexical relation that holds for all syntactic categories but is essential in the formation of adjective clusters. For each antonymous pair, both relations are listed (i.e. each *synset_id,w_num* pair is both a source and a target word).

When studying the contents of file **wn_ant.pl** of the WN Prolog database, which contains all Prolog facts referring to antonymous words, one easily notices that the great majority of these facts establish direct antonymy relations among words occurring as first elements within the synsets to which they belong. Less than 15 exceptions to this rule exist. These exceptions can be easily processed by a human operator retaining the new positions of the adjectives having direct antonyms. Under these circumstances, we have found it justifiable to formulate

Remark 3.1

The first word of an English adjective synset is the one possibly having a direct antonym.

Let us assume that all translated (foreign) adjective synsets exist and that they belong to a file named **wn_strans.pl**. Using Remark 3.1 and having generated file **wn_strans.pl** by applying the translation algorithm, we can now formulate the algorithm for generating the foreign adjective clusters corresponding to the English ones:

Algorithm 3.1

Input: Files **wn_ant.pl, wn_sim.pl**, and **wn_strans.pl**

For each *synset pair* denoted by each Prolog fact of file **wn_ant.pl** perform steps 1. to 5.:

1. Look in file **wn_strans.pl** and find the foreign synsets representing the translations of the considered English ones.

2. Corresponding to each foreign synset found in **wn_strans.pl** in step 1. retain the first word of that synset. (This word pair will be used in the foreign cluster head.)

3. For the same word pair look in file **wn_sim.pl** and take into consideration the **sim** clauses corresponding to each of the two synsets to which the two words of the cluster head belong.

4. Take into account all synsets denoted by the **sim** clauses chosen in step 3., synsets having the second *id* which occurs in the clause. Find the foreign synsets representing their translations in file **wn_strans.pl**.

5. Add each first word of these foreign synsets in the cluster head, together with the & pointer.

6. Add each "similar" foreign synset, ending it with the reciprocal similarity pointer.

Output: A file containing all foreign adjective clusters.

Algorithm 3.1 will generate foreign adjective clusters with a bipolar structure like the one described in §3.1 and illustrated in Figure1.

At this early stage of our study we have been concerned uniquely with creating the *WN type* cluster structure and have not tried to distinguish among different subsenses or different privileges of occurrence. We have equally not tried to indicate the limitation of certain adjectives as to the syntactic positions they can occupy, a word-form limitation which in WN is coded for individual adjectives. This can easily be achieved once the basic algorithm has been established. Other issues, such as the capitalized pointers sometimes occurring in the structure of WN clusters, which serve as "see also" cross-references to related clusters, have also been ignored for the time being. All these and others represent topics for future study.

Obviously, according to the chosen target language, various difficulties of linguistic nature will be encountered. For instance, identical foreign synsets might be generated by Algorithm 3.1 corresponding to different English ones, namely to different meanings and concepts. This is the case when an English polysemous adjective will have one or more meanings in English that do not exist in the target language, a situation which is called *semantic loan*, leading to *loan translation.* Linguistic validation of the output of computer programs implementing Algorithm 3.1, or any other algorithm of the same type, for that matter, will always be necessary. However, we consider that Algorithm 3.1 accounts for the great majority of cases when dealing with adjective clusters of WN type.

4 Final Remarks

Our paper is an attempt to discuss the semiautomatic generation of WNs for languages other than English by means of a class method, namely a method that uses as knowledge sources individual entries coming from bilingual dictionaries and WN synsets.

The topic itself is of great interest since these WNs will create an infrastructure for developing knowledge processing systems such as automatic translation between English and other languages, information retrieval and extraction from documents in languages for which WNs exist, and many other advanced Information Technology systems.

The proposed approach to semiautomatic WN generation is a combination of automatic and manual methods. The manual method relies on human experts, while the automatic class method relies uniquely on bilingual dictionaries.

Using the proposed class method (which is language independent and irrespective of part of speech) is sufficient in order to automatically generate the synsets of the target language (which will be manually validated). In the case of adjectives, however, one should be concerned not only with automatically translating English adjective synsets but also with creating the typical adjective cluster structure corresponding to the target language. In order to achieve this the WN structure should be taken into account, a fact which denotes the necessity of combining class methods with structural ones. The generation of adjective clusters can be accomplished entirely automatically (using Algorithm 3.1), provided that the translation of the involved adjective synsets, performed by means of the proposed class method, has been validated by human experts.

The significance of the manual effort involved in quality assurance primarily depends on the existence of appropriate tools. It is our belief that the involved human effort is greatly reduced in the case of those languages for which correct and complete bilingual dictionaries in electronic format exist.

Acknowledgements

1. The present study has been elaborated within the framework of the BALRIC-LING research project, as part of the IST Programme, funded by the European Commission (Contract No: IST-2000-26454). The author would like to thank the European Commission for the importance it has attached to the presented topic, as well as for having offered its full support.

2. The author is greatly indebted to four anonymous referees for their extremely valuable comments and suggestions.

References

[Atserias et. al., 97] Atserias, J., Climent, S., Farreres, X., Rigau, G., Rodriguez, H.: "Combining Multiple Methods for the Automatic Construction of Multi-lingual WordNets"; in: Recent Advances in Natural Language Processing II. Selected papers from RANLP'97. Edited by Nicolas Nicolov and Ruslan Mitkov; John Benjamins Publishing Company, Amsterdam / Philadelphia (1997), 327 - 338.

[Fellbaum, 98] Fellbaum, C.(Ed.): "WordNet: An Electronic Lexical Database"; The MIT Press, Cambridge/London/England(1998).

[Gross et. al., 89] Gross, D., Fisher, U., Miller, G.A.: "The organization of adjectival meanings"; Journal of Memory and Language, 28 (1989), 92 - 106.

[Levi, 78] Levi, J.N.: "The Syntax and Semantics of Complex Nominals"; Academic Press, New York (1978).

[Leviţchi, 73] Leviţchi, L.: "Dicţionar român-englez" (3rd edition); Editura Ştiinţifică, Bucureşti (1973).

[Leviţchi et. al., 74] Leviţchi, L., Bantaş, A., Nicolescu, A.: "Dicţionar englez-român"; Editura Academiei Române, Bucureşti (1974).

[Miller et. al., 90] Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.J.: "Introduction to WordNet: an on-line lexical database"; International Journal of Lexicography, 3,4(1990), 235-244.

[Nikolov and Petrova, 00] Nikolov, T., Petrova, K.: "Building and Evaluating a Core of Bulgarian WordNet for Nouns"; OntoLex '2000 Report, Sozopol, Bulgaria (2000).

[Nikolov and Petrova, 01] Nikolov, T., Petrova, K.: "Towards Building Bulgarian WordNet"; Proc. RANLP'01, INCOMA Ltd., Tzigov Chark, Bulgaria (2001), 199-203.