On the Semiautomatic Generation of WordNet Type Synsets and Clusters with Special Reference to Romanian

by

Florentina Hristea

1 Introduction. What is WordNet

WordNet 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, WordNet being organized according to semantic relations which are indicated by pointers between synsets.

WordNet 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, WordNet 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*, WordNet is much more useful to Artificial Intelligence applications since it is enriched with an impressive set of relations among words and word meanings. WordNet distinguishes between semantic relations and lexical relations, but the emphasis is on semantic relations between meanings. Therefore, unlike standard dictionaries, WordNet organizes lexical information in terms of word meanings, rather than word forms. WordNet 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 WordNet 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 WordNet. 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), WordNet 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 WordNet continues to grow. Version 1.7 is now accessible as well, at

http://www.cogsci.princeton.edu/~wn/obtain/

The most ambitious feature of WordNet is most probably the semantic attempt and, in this respect, WordNet 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 WordNet 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 WordNet 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 WordNet "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.

WordNet 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 WordNets 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** that will 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 WordNet.

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 meaning 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 meaning 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 the following

sequence be an EW, while *fword1, fword2* and *fword3* are its corresponding translation equivalents (according to the appropriate bilingual dictionary):

eword *fword1*; *fword2*, *fword3*

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

{*fword1*} and {*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 below.

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 at 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:

eword1 meaning11; meaning12; $\cdots$; 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 $meaningji_j$.

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:

$$\{meaningli_1\} \cup \{meaning2i_2\} \cup \ldots \{meaningni_n\}, 1 \leq i_j \leq m_j,$$

$$\forall\, j = \overline{1,n}.$$

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.

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.* All tests performed have been using the original WordNet 1.6 in its Prolog-readable format. In

order to test the algorithms we have used fragments of bilingual dictionaries since complete Romanian-English and English-Romanian dictionaries in electronic format were not available. 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.

The fragments of electronic bilingual dictionaries used in the experiment were created by us, using [DRE, 73] and [DER, 74]. Our electronic bilingual dictionaries contained a total of 1164 words, out of which 278 were English nouns (corresponding to the 200 studied English synsets) and 886 words were the corresponding Romanian nouns representing their translations. The files containing these bilingual dictionaries are part of the input for the computer program implementing Algorithm 2.1. The Prolog format of the dictionaries, used in our Prolog implementation of all algorithms, can be seen in §3.2.2. The generated noun Romanian synsets corresponding to the 200 studied English ones and representing the output of Algorithm 2.2 can be seen on the web[1]. The generated Romanian synsets were validated by Romanian linguists using the latest bilingual dictionaries and the corresponding gloss indicated in the American WordNet. 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.

[1]http://phobos.cs.unibuc.ro/roric/topic2.html

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 almost 96% of the considered English synsets the generated Romanian ones were correct. The remaining ones did not have correct Romanian counterparts 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, providing that good bilingual dictionaries in electronic format exist for the specific foreign language involved.

The greatest advantage of the proposed translation algorithm 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. When testing a variant of the presented translation algorithm for *Bulgarian noun synsets*, for instance, phenomena like the lack of a regular conversion in Bulgarian, the translation of a gerund by a deverbial noun or by a special type of an infinitive or a subordinate clause, the existence of rich systems of participles and others are taken into account in [Nikolov and Petrova, 01]. Reported results are however also very good.

A general difficulty, of a different nature, encountered no matter what language is taken into consideration, consists of what we might call the cross-language wide meaning of a given word. Namely, a word in one language sometimes covers a relatively wide concept and is connected to more than one word in another language where each of the words it is linked to describes a more specific concept. This is a very important issue from the WordNet approach point of view, since in WordNet synsets exist according to the corresponding underlying concepts.

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 WordNet from the original WordNet 1.6. 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

WordNet 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. WordNet 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 WordNet this binary opposition is represented by reciprocal labeled pointers: *heavy!*→*light* and *light!*→ *heavy.* In the Prolog implementation of WordNet, 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 WordNet 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, Fisher 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 WordNet, 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.*

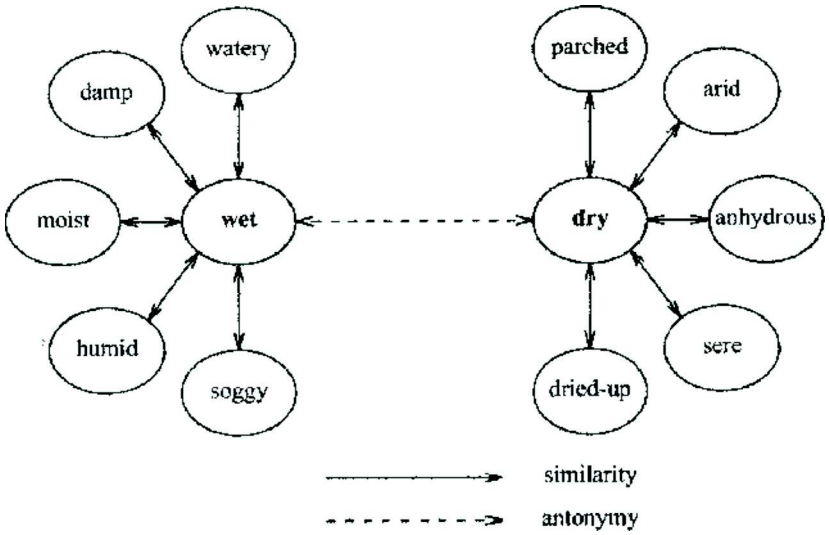

Figure 1. Bipolar Adjective Structure

Corresponding to Figure 1, which intuitively presents the structure of an adjective cluster in WordNet, 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 WordNet, that the presented model accounts for the great majority of the English descriptive adjectives. The importance of the similarity relation is obvious.

In WordNet 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 WordNet database, the *synset_id*s are used as unique synset identifiers. As it was already noted, in the Prolog implementation of WordNet 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 WordNet 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 can not be incorporated into the clusters that characterize descriptive adjectives. WordNet 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

3.2.1 The algorithm

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 WordNet 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.

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 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

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

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.

3.2.2 Prolog implementation

The WordNet database in a Prolog-readable format is contained in the files **wn_⋆.pl** and uses synset *id*s as unique synset identifiers. A separate file has been created for each WordNet relation giving the user the ability to load only those parts of this very large database that they are interested in. Each Prolog database file contains information corresponding to the synsets and word senses contained in the WordNet database. Each line of a file contains an operator that corresponds to a WordNet relation. All lines with the same *operator* value are stored in the file **wn_***operator.pl.* The general format of a line in a Prolog database file is

$$operator(field1, \ldots, fieldn).$$

containing no spaces and being terminated with a newline character.

An **s** operator is present for every word sense in WordNet, while the **g** operator specifies the gloss for a synset. File **wn_s.pl** contains all WordNet

synsets, while file **wn_g.pl** contains all glosses.

File **wn_s.pl** contains all WordNet synsets and is formed of Prolog facts having the general form

s(*synset_id,w_num,'word',ss_type,sense_number,tag_state).*

where:
- *synset_id* represents the synset identifier;
- *w_num* is the word number within the synset;
- '*word*' is the ASCII text of the actual word to which the Prolog fact refers;
- *ss_type* is a one character code indicating the type of the synset which designates the part of speech of 'word'. (In the case of adjectives this type can be *a*, standing for adjective, or *s*, standing for adjective_satellite);
- *sense_number* is a natural number denoting which sense of the word the synset refers to; sense ordering is performed according to the tag_state;
- *tag_state* can be 1 or 0, having the significance

  1 - the senses of the word are ordered according to frequency of use

  0 - the senses of the word are not ordered based on frequency of use, but according to other criteria.

An **s** operator is present for every word sense in WordNet.

An example of such a Prolog fact, corresponding to the English adjective *abridged*, is the following:

s(300004481,1,'abridged',a,1,0).

Due to the dimensions of the WordNet database, in order to decrease computer time, file **wn_s.pl** has been divided by us into smaller files containing all Prolog facts referring to the same part of speech. This file is part of the input of all programs dealing with the automatic generation of foreign synsets.

In order to decrease computer time even more, while eliminating all information which is unnecessary for our present task, file **wn_s.pl** containing only Prolog facts corresponding to adjectives has been processed even further. Computer programs have been written in order to obtain a simplified form of the Prolog facts referring to adjectives.

First of all, parameters a and/or s, denoting the part of speech, have been eliminated. The second and last two parameters have also been found unnecessary for our present purpose. Capital letters and special symbols like apostrophes have equally been eliminated, as well as Prolog facts containing proper nouns and collocations. The latter should be subject to a separate study. The resulting file containing adjective synsets includes Prolog facts of the following form:

$$s(300003469,'emergent').$$
$$s(300003469,'emerging').$$
$$s(300003469,'nascent').$$

Finally, the resulting file has been processed such that a unique Prolog fact was obtained corresponding to each synset. In the case of our present example, the following Prolog fact was created:

s(300003469, [emergent, emerging, nascent]).

Note that, when appending the words in order to create the synset, it is essential to maintain their order of occurrence due to our decision to always choose the adjective placed on the first position when enriching synsets containing a unique word via the similarity relation. The way in which adjective clusters are built also makes this necessary. The file containing Prolog facts of the above form, one fact corresponding to each adjective synset in WordNet, is part of the input of all computer programs implementing both Algorithm 2.1 and Algorithm 2.2.

We shall refer to this file as being "*the cleared, combined synset file*". Such cleared, combined synset files have been used both in the case of nouns and in the case of adjectives, when implementing the proposed algorithms. The present comments refer mainly to adjectives since processing adjective synsets required some additional steps beyond obtaining this type of file.

The cleared, combined synset file is part of the input of all computer programs implementing Algorithm 2.1 and Algorithm 2.2 when *translation without enrichment* is required.

Another important part of the input of all computer programs are the two bilingual dictionaries. In the case of the example we referred to in § 3.2.1 (the word *sticky*), a dictionary entry in the English-Romanian dictionary is of the form

sticky lipicios , cleios , vascos ; umed , cetos

We have been using only two separators, the comma and the semicolon. While a comma separates synonyms referring to the same meaning of the word, the semicolon is used as a separator between meanings. Each separator is preceded and followed by a blanc. The English word is the only one followed by more than one blanks. Corresponding to such a dictionary entry, the following Prolog fact has been obtained:

word1(sticky, [[lipicios, cleios, vascos], [umed, cetos]]).

The above predicate (*word1)* has two parameters, namely the English word and the list of corresponding e-sets. This list is included in square brackets. Each e-set is included in square brackets as well. The Romanian-English dictionary consists of similar Prolog facts.

On the web[1] you will find **the Java implementation** of all proposed

[1]http://phobos.cs.unibuc.ro/roric/topic2.html

algorithms, which uses similar formats of the bilingual dictionaries, as will be described there.

The next step in preparing the implementation of Algorithm 2.1 consists of automatically enriching those adjective synsets containing just one word by means of the similarity relation. The computer program which performs this operation receives as input the cleared, combined synset file and file **wn_sim.pl** of the WordNet Prolog database, which contains the Prolog facts corresponding to the existing similarity relations. The output is a file containing Prolog facts of the form

st1(synset_id,synset).

This file includes all adjective synsets in WordNet, with those having contained just one word being enriched. This is, in fact, **the final synset file**, representing the input for the computer programs implementing Algorithm 2.1 and Algorithm 2.2 when *translation with enrichment* is required.

As far as the Prolog implementation of semantic and lexical relations in WordNet is concerned, the issue is discussed in § 3.1 and § 3.3, when referring to the **sim** and to the **ant** operators respectively.

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 WordNet 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:

$$\textbf{ant}(synset_id,w_num,synset_id,w_num).$$

Thus, the significance of the following Prolog fact

$$\textbf{ant}(302425348, 1, 302429323, 1).$$

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 WordNet 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** at 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 at 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 Figure 1. Corresponding to this cluster, namely the one having the antonym pair [WET, DRY, !] and [DRY, WET, !] respectively in the head synset, the following Romanian cluster has been generated:

[{ [UMED, USCAT, !] inrourat,& stropit,& vascos,& umed,& umed,& cetos,& lipicios,& ploios,& lipicios,& umed,& }
{ inrourat, stropit, umezit, umed,& }
{ stropit, smaltat, umed,& }
{ vascos, cleios, lipicios, umed,& }
{ umed, igrasios, jilav, ud, umed,& }
{ umed, jilav, ud, umed,& }
{ cetos, aburit, umed, jilav, ud, umed,& }
{ lipicios, cleios, vascos, umed,& }
{ ploios, umed,& }
{ lipicios, cleios, vascos, umed, jilav, ud, umed,& }
{ umed, jilav, ud, umed,& }
-
{ [USCAT, UMED, !] arid,& uscat,& secat,& uscat,& uscat,& uscat,& }
{ arid, uscat, sec, uscat,& }
{ uscat, arid, sterp, sec, uscat,& }
{ secat, uscat,& }
{ uscat, ofilit, vestejit, zbarcit, uscat,& }
{ uscat, arid, sterp, sec, uscat,& }
{ uscat, arid, insetat, uscat,& }]

Note that not all existing similarity relations have been used in this case, since our aim was only that of offering an example of the *type* of structure which is generated by Algorithm 3.1.

Our Romanian cluster only illustrates the basic coding devices used in the American WordNet, which we have also made use of. At this early stage of our study we have been concerned uniquely with creating the *WordNet 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 WordNet 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 WordNet 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 WordNet type.

4 Final Remarks

WordNet has been recognized as a valuable resource in the human language technology and knowledge processing communities. Many researchers who use WordNet especially in the field of Artificial Intelligence view it

primarily as a lexical *knowledge base* and make subsequent use of it. Knowledge processing has gained new dimensions in the U.S. due to the existence of WordNet. Its applicability has been cited in more than 200 papers and systems have been implemented using it. Many groups of researchers expressed their interest in WordNet applications in various fields, such as : Information Retrieval, Information Extraction, Word Sense Disambiguation, Text Inference, Natural Language Generation, Learning, Knowledge Acquisition and others.

The human language research community has encouraged the development of WordNets 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* that will simplify translation from a language to another and will facilitate the use of the same reasoning schemes and algorithms already developed in conjunction with the American WordNet.

Acknowledgments

1. The author would like to thank Prof. Dr. Theodor Hristea of the Faculty of Letters, the University of Bucharest, for having provided constant linguistic guidance concerning WordNet, as well as for the validation of Romanian generated synsets and clusters.

2. The author would like to thank Master of Science students Cristina Vaţă, Claudia Burtea and Mihai Sima of the Faculty of Mathematics, the University of Bucharest, as well as Raluca Elena Gălăţanu of the Faculty of Letters of the same university, for having assisted the RORIC-LING team, with much dedication, during this second phase of the BALRIC-LING project.

References

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

[Harabagiu, 99] Harabagiu, S.: "Lexical Acquisition for a Romanian WordNet"; Proc. EUROLAN '99, Iaşi, Romania (1999).

[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.

[DRE, 73] Levitchi, L., "Dicţionar român-englez" (ed.a III-a); Editura Ştiinţifică, Bucureşti (1973).

[DER, 74] "Dicţionar englez-român"; Editura Academiei R.S.R., Bucureşti (1974).