A Complete WN1.5 to WN1.6 Mapping

J. Daudé, L. Padró and **G. Rigau**
TALP Research Center
Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya. Barcelona
{daude,padro,g.rigau}@lsi.upc.es

Abstract

We describe a robust approach for linking already existing lexical/semantic hierarchies. We use a constraint satisfaction algorithm (relaxation labelling) to select –among a set of candidates– the node in a target taxonomy that bests matches each node in a source taxonomy. In this paper we present the complete mapping of the nominal, verbal, adjectival and adverbial parts of WordNet 1.5 onto WordNet 1.6.

1 Introduction

There is an increasing need of having available general, accurate and broad coverage multilingual lexical/semantic resources for developing NL applications. Thus, a very active field inside NL during the last years has been the fast development of generic language resources.

Several attempts have been performed to connect already existing ontologies. For instance, to (semi)automatically link Spanish taxonomies extracted from DGILE (Alvar, 1987) to their English analogous from LDOCE (Ageno et al., 1994), or to WordNet (Miller et al., 1991) synsets (Rigau et al., 1995); the construction of large multilingual lexicons as in (Knight and Luk, 1994; Okumura and Hovy, 1994); or the alignment of EDR and WordNet ontologies (Utiyama and Hasida, 1997). Also, several lexical resources and techniques are combined in (Atserias et al., 1997; Farreres et al., 1998) to map Spanish words from a bilingual dictionary to WordNet .

The use of relaxation labelling algorithm to attach substantial fragments of the Spanish taxonomy derived from DGILE (Rigau et al., 1998) to the English WordNet has been reported in (Daudé et al., 1999). The mapping of the nominal parts of WN1.5 to WN1.6 is presented in (Daudé et al., 2000).

In this paper we use the same approach to map the verbal, adjectival and adverbial parts of WN1.5 to WN1.6, and present some improvements on the results presented in (Daudé et al., 2000). We provide evaluation of the mapping accuracy by means of hand checking a random sample of the taxonomies.

This paper is organized as follows: In section 2 we describe the used technique (the relaxation labelling algorithm) and its application to hierarchy mapping. In section 3 we describe the constraints used in the relaxation process, and finally, section 4 details performed tests and evaluation of achieved results.

2 Application of Relaxation Labelling to NLP

Relaxation labelling (RL) is a generic name for a family of iterative algorithms which perform function optimization, based on local information, but with global effects. See (Torras, 1989) for a summary. Its most remarkable feature is that it can deal with any kind of constraints, and the algorithm is independent of the complexity of the model.

The algorithm has been applied to POS tagging (Màrquez and Padró, 1997), shallow parsing (Voutilainen and Padró, 1997) and to word sense disambiguation (Padró, 1998).

2.1 Algorithm Description

The Relaxation Labelling algorithm deals with a set of variables (which may represent words, synsets, etc.), each of which may take one among several different labels (POS tags, senses, MRD entries, etc.). There is also a set of constraints which state compatibility or incompatibility of a combination of pairs variable–label.

The aim of the algorithm is to find a weight assignment for each possible label for each vari-

able, such that (a) the weights for the labels of the same variable add up to one, and (b) the weight assignment satisfies –to the maximum possible extent– the set of constraints.

Summarizing, the algorithm performs constraint satisfaction to solve a consistent labelling problem. The followed steps are:

1. Start with a random weight assignment.
2. Compute the *support* value for each label of each variable. Support is computed according to the constraint set and to the current weights for labels belonging to context variables.
3. Increase the weights of the labels more compatible with the context (larger support) and decrease those of the less compatible labels (smaller support). Weights are changed proportionally to the support received from the context.
4. If a stopping/convergence criterion is satisfied, stop, otherwise go to step 2. We use the criterion of stopping when there are no more changes, although stopping criteria may range from simply a fixed number of iterations to more sophisticated heuristic procedures (Eklundh and Rosenfeld, 1978).

2.2 Application to taxonomy mapping

As described in previous sections, the problem we are dealing with is to map two taxonomies. In this particular case, we are interested in mapping WN1.5 to WN1.6, that is, assign each synset in the former to at least one synset in the later.

The modeling of the problem is the following:

- Each WN1.5 synset is a variable for the relaxation algorithm. We will refer to it as *source synset* and to WN1.5 as *source taxonomy*.
- The possible labels for that variable are all the WN1.6 synsets which contain a word belonging to the source synset. We will refer to them as *target synsets* and to WN1.6 as *target taxonomy*.
- The algorithm will need constraints stating whether a WN1.6 synset is a suitable assignment for a WN1.5 synset. As described in section 3, these constraints will rely mainly on the taxonomy structure.

3 The Constraints

Constraints are used by the relaxation labelling algorithm to increase or decrease the weight for a variable label. In our case, constraints increase the weights for the connections between a source synset and a target synset. Increasing the weight for a connection implies decreasing the weights for all the other possible connections for the same source synset. To increase the weight for a connection, constraints take into account already connected nodes that have the same relationships in both taxonomies.

3.1 Constraints for WN Nouns

Although there is a wide range of relationships between WordNet synsets which can be used to build constraints, in (Daudé et al., 2000) we focus on the hyper/hyponym relationships, increasing the weight for a connection when the involved nodes have hypernyms/hyponyms also connected. Hyper/hyponym relationships are considered either directly or indirectly (i.e. ancestors or descendants), depending on the kind of constraint used.

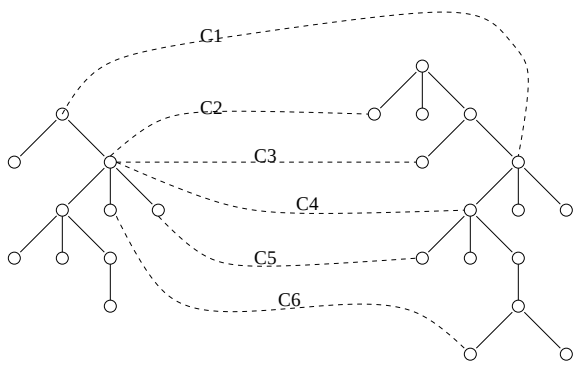

Figure 1: Example of connections between taxonomies.

Figure 1 shows an example of possible connections between two taxonomies. Connection C_4 will have its weight increased due to C_5, C_6 and C_1, while connections C_2 and C_3 will have their weights decreased.

In (Daudé et al., 2000) we distinguish different kinds of constraints, depending on whether hyponyms, hypernyms or both are taken into account, on whether those relationships are considered direct or indirect, and on which of both taxonomies recursion is allowed. Each constraint may be used alone or combined with others.

Thus, the constraints used in (Daudé et al., 2000) are:

II **constraints.** This group includes IIE, IIO and IIB constraints. All of them match an immediate–to–immediate relationship between both taxonomies, as shown in figure 2, where the arrows indicate an immediate hypernymy relationship. The nodes on the left hand side correspond to the source taxonomy and the nodes on the right to the target hierarchy. The dotted line is the connection which weight will be increased due to the existence of the connection(s) indicated with a continuous line.

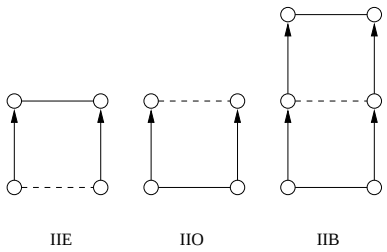

Figure 2: II constraints.

AI **constraints.** This constraint group includes AIE, AIO and AIB. That is, constraints that allow recursion on the source taxonomy. A graphical representation is shown in figure 3, where the + sign indicates that the hypernymy relationship represented by the arrow does not need to be immediate. In this case, this iteration is only allowed in the source taxonomy.

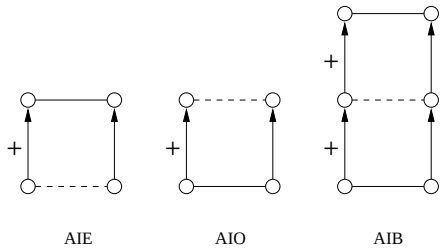

Figure 3: AI constraints.

IA **constraints:** Are symmetrical to AI constraints. In this case, recursion is allowed only on the target taxonomy.

AA **constraints:** Include the same combinations than above, but allowing recursion on both sides, as presented in figure 4.

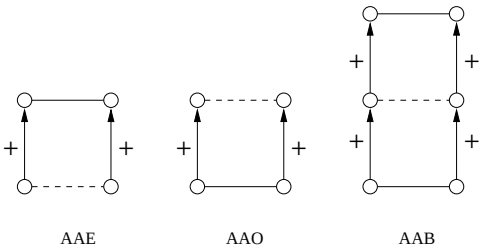

Figure 4: AA constraints.

3.2 Generalized Constraints

Although the minimalist approach described above yields very good results for the WordNet nominal taxonomy using only hyper/hyponymy relationships, it becomes much less useful in the case of verbs –due to the very flat verb taxonomy in WN– and completely useless for adjectives and adverbs, which do not have such a kind of relationships.

Thus, we generalized the structural matching idea to relationships other than hyper/hyponyms. In that way, we can use constraints that involve any WN relationship such as antonymy, "also-see", "similar-to", etc.

This generalized constraint class has the schema presented in figure 5, where a connection between a source synset s_1 and a target synset t_1 is reinforced if there are a source synset $s2$ and a target synset $t2$ such that $s_1 \mathcal{R} s_2$ and $t_1 \mathcal{R} t_2$, where $\mathcal{R}$ may be any WN relationship.

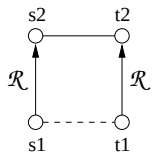

Figure 5: Generalized constraints.

This new scenario enables us to map the complete WN1.5 to WN1.6. Now we can talk about source and target *graphs*, since it is not only hyper/hyponymy what defines the structure which constraints the mapping.

Nevertheless, this graph mapping is performed incrementally for efficiency reasons, as described in section 3.4.

3.3 Additional Heuristic Constraints

Apart from the generalized structural constraints presented in section 3.2, some heuristic, non-structural constraints were also used to

help the algorithm decide in some cases in which structure was not enough informative. For instance, the case where a leaf node in WN1.5 may map to two different leafs in WN1.6, both under the same parent, may be disambiguated successfully with some simple heuristic constraints.

A rough measure of similarity between two synsets can be computed using the number of word coincidences, the number of non–empty word coincidences in glosses, or –in the case of verb synsets– the number of frame coincidences.

For each of these similarity measures, a non-structural constraint can be stated, which reinforces a connection between a source and a target synset proportionally to their similarity.

We denote these constraints with W, G and F, depending on whether word, gloss or frame coincidences are used to compute similarity.

3.4 Constraints for WN Verbs, Adjectives and Adverbs

In order to reduce the computation time for the mapping, we proceeded in three phases, following the relationships that establish dependences among different POS in WordNet, as described below:

- First, we mapped nouns, using only hyper/hyponymy relationships plus W and G additional constraints.

  Similarly, verbs were mapped using hyper/hyponymy, antonymy and the "also-see" WN relationship. W, G and F additional constraints were also used.

- Second, adjectives were mapped using *adj–to–adj* relationships such as antonymy, "similar-to" and "also-see", as well as the *adj–to–verb* relationship "participle-of" and the *adj–to–noun* "pertains" and "attribute". W and G additional constraints were also used.

  Notice that the graph used to check the constraints imposed by *adj–to–verb* and *adj–to–noun* relationships was the result of the previous steps. That means that verbs and nouns are already mapped, reducing considerably the search space and accelerating relaxation labelling convergence. Since the noun and verb mappings do not depend on adjectives (there are no *noun–to–adj* relationships and only one *verb–to–adj* –not used to avoid circularity–), results wouldn't have been greatly affected if the mappings had been performed in parallel, but the convergence would have been slower.

- The third and last phase is the adverb mapping, which is performed with the only *adv–to–adv* relationship, antonymy and with the *adv–to–adj* "derived". W and G additional constraints were also used.

  Again, the noun, verb, and adjective graphs are taken as a static picture of the mapping obtained from previous phases.

4 Experiments and Results

In the performed tests we evaluated the performance of different constraint combinations for each phase.

All figures presented in this section were computed by manually linking to WN1.6 a sample chosen from WN1.5, and then use this sample mapping as a reference. The validation sample consists of 1900 noun synsets, 1000 verb, 1000 adjective and 300 adverb synsets.

See (Daudé et al., 2000) for a comparison of our mapping with the *SenseMap* provided by Princeton[1].

Results are presented in tables 1, 2 and 4. They present the results for increasingly complex constraint sets.

First, table 1 presents results with the *basic* constraint set for each part–of–speech. They consist of the hyper/hyponym relationships (when available) plus other available *intra*-POS relationships –i.e. relationships between synsets with the same POS. Namely, the constraint sets are:

Nouns: AA hyper/hyponym constraint set – corresponding to the basic technique used in (Daudé et al., 2000).

Verbs: AA hyper/hyponym constraint set plus "also-see" and antonymy relationships in II (immediate–to–immediate) mode.

Adjectives: Antonymy, "similar-to" and "also-see", all of them in II mode.

Adverbs: Antonymy in II mode.

[1] See WN web page at http://www.cogsci.princeton.edu/~wn/

	Cover.	ambiguous.	overall
N	99.7%	94.9%–99.6%	97.6%–99.8%
V	96.9%	93.5%–99.2%	94.6%–99.2%
A	94.1%	82.8%–98.9%	89.5%–99.4%
R	80.8%	97.5%–100.0%	99.0%–100.0%

Table 1: Precision–recall results for *basic* constraint set.

Second, table 2 shows the results for the *basic* set extended with the W and G constraints for nouns, adjectives and adverbs, and with W, G and F for verbs.

Note that the results for verbs are lower over all synsets than over ambiguous ones. This is caused because some of the non–ambiguous WN1.5 synsets in the validation sample were hand marked to have no correspondence in WN1.6 (i.e. they were removed in the new version), which causes the precision for non–ambiguous nodes to be only 99.1%.

	Cover.	ambiguous.	overall
N	99.9%	97.5%–97.7%	98.8%–98.9%
V	99.8%	99.4%–99.7%	99.3%–99.6%
A	98.9%	96.5%–98.8%	97.9%–99.3%
R	99.5%	97.5%–100.0%	99.0%–100.0%

Table 2: Precision–recall for *basic*+WGF constraint set

Third, table 3 shows the results for the *basic* set extended with available *extra*–POS relationships, which are:

Nouns: No *extra*–POS relationships.

Verbs: No *extra*–POS relationships.

Adjectives: "participle-of", "attribute" (*adj–to–noun*) and "pertains-to" (*adj–to–verb*), both in II mode.

Adverbs: "derived–from" (*adv–to–adj*), in II mode.

	Cover.	ambiguous.	overall
N	–	–	–
V	–	–	–
A	95.8%	85.0%–98.9%	90.9%–99.4%
R	88.0%	69.2%–94.2%	87.9%–98.1%

Table 3: Precision–recall for *basic*+*extra*–POS constraint set

Finally, Table 4 shows the results obtained when using together the *basic* set, plus the WGF set, plus the *extra*-POS set. For nouns and verbs, results from table 2 are repeated for clarity.

	Cover.	ambiguous.	overall
N	99.9%	97.5%–97.7%	98.8%–98.9%
V	99.8%	99.4%–99.7%	99.3%–99.6%
A	99.0%	96.5%–99.1%	97.9%–99.5%
R	99.6%	98.3%–100.0%	99.3%–100.0%

Table 4: Precision–recall for complete constraint set.

Those figures indicate that in all cases, the use of additional constraints other than the *basic* set, produces an increase in coverage.

For the WGF constraints, this improvement is accompanied by a large ambiguity reduction (thus a narrower precision–recall interval) and an increment in precision, specially in verbs and adjectives. Adverbs increase coverage maintaining precision and recall, while nouns present a small improvement.

The *extra*-POS constraints turned to be useful for adjectives, but harmful for adverbs when used alone. In both cases, they raised precision when used in conjunction with WGF constraints.

The recall below 100% is caused by synsets having several candidates with the same structural features. Since all candidates obtain the same support from the constraints, the algorithm cannot chose one or another and leaves it ambiguous. If the final support is under the threshold, no answer is computed as correct.

The precision under 100% is caused by two kinds of errors: First, some WN1.5 synsets are splitted in WN1.6 and the human evaluator may consider that the best choice is not the one made by the system. Second, some WN1.5 synsets simply do not appear in WN1.6, but if the system assigns them anyway, they are computed as an error.

5 Conclusions & Further Work

We have applied the relaxation labelling algorithm to build a complete mapping between two different WordNet versions, and reported results on WN1.5 to WN1.6 complete mapping[2].

[2]The mapping has been made public at http://www.lsi.upc.es/~nlp. So far, 37 licenses have been provided.

An incremental approach has been used to map separately the different WN POS files, using at each step the results of previous ones.

Non–structural information such as synset words, glosses and verb frames has been proven useful in increasing coverage, reducing ambiguity, and improving performance.

6 Acknowledgments

This research has been partially funded by the the UE Commission (NAMIC IST-1999-12392), by the Spanish Research Department (TIC98-423-C06-06, TIC2000-1735-C02-02) and by the Catalan Research Department, (Quality Research Group Programme, GRQ93-3015).

References

A. Ageno, I. Castellón, F. Ribas, G. Rigau, H. Rodríguez, and A. Samiotou. 1994. TGE: Tlink Generation Environment. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING'94)*, Kyoto, Japan.

M. Alvar, editor. 1987. *Diccionario General Ilustrado de la Lengua Española VOX*. Bibliograf S.A, Barcelona, Spain.

J. Atserias, S. Climent, X. Farreres, G. Rigau, and H. Rodríguez. 1997. Combining Multiple Methods for the Automatic Construction of Multilingual WordNets. In *proceedings of International Conference on Recent Advances in Natural Language Processing (RANLP'97)*, Tzigov Chark, Bulgaria.

J. Daudé, L. Padró, and G. Rigau. 1999. Mapping Multilingual Hierarchies Using Relaxation Labeling. In *Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC'99).*, Maryland, US.

J. Daudé, L. Padró, and G. Rigau. 2000. Mapping WordNets Using Structural Information. In *38th Annual Meeting of the Association for Computational Linguistics (ACL'2000).*, Hong Kong.

J. O. Eklundh and A. Rosenfeld. 1978. Convergence Properties of Relaxation Labelling. Technical Report 701, Computer Science Center. University of Maryland.

X. Farreres, G. Rigau, and H. Rodríguez. 1998. Using WordNet for Building WordNets. In *Proceedings of COLING-ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, Montréal, Canada.

K. Knight and S. Luk. 1994. Building a Large-Scale Knowledge Base for Machine Translation. In *Proceedings of the American Association for Artificial Inteligence (AAAI'94).*

L. Màrquez and L. Padró. 1997. A Flexible POS Tagger Using an Automatically Acquired Language Model. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics. Joint ACL/EACL*, pages 238–245, Madrid, Spain, July.

G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. 1991. Five Papers on WordNet. *International Journal of Lexicography.*

A. Okumura and E. Hovy. 1994. Building japanese-english dictionary based on ontology for machine translation. In *proceedings of ARPA Workshop on Human Language Technology*, pages 236–241.

L. Padró. 1998. *A Hybrid Environment for Syntax–Semantic Tagging.* Phd. Thesis, Dep. Llenguatges i Sistemes Informàtics. Universitat Politècnica de Catalunya, February. http://www.lsi.upc.es/~padro.

G. Rigau, H. Rodríguez, and J. Turmo. 1995. Automatically extracting Translation Links using a wide coverage semantic taxonomy. In *proceedings 15th International Conference AI'95*, Montpellier, France.

G. Rigau, H. Rodríguez, and E. Agirre. 1998. Building Accurate Semantic Taxonomies from MRDs. In *Proceedings of COLING-ACL'98*, Montréal, Canada.

C. Torras. 1989. Relaxation and Neural Learning: Points of Convergence and Divergence. *Journal of Parallel and Distributed Computing*, 6:217–244.

M. Utiyama and K. Hasida. 1997. Bottom-up Alignment of Ontologies. In *Proceedings of IJCAI workshop on Ontologies and Multilingual NLP*, Nagoya, Japan.

A. Voutilainen and L. Padró. 1997. Developing a Hybrid NP Parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing, ANLP*, pages 80–87, Washington DC. ACL.