Multilingual lexical representation

Ann Copestake, Bernie Jones, Antonio Sanfilippo

University of Cambridge Computer Laboratory
New Museums Site, Pembroke Street, Cambridge, CB2 3QG, UK

Horacio Rodríguez

Dept. Llenguatges i Sistemes Informatics, Universitat Politècnica de Cataluyna
PAU GARGALLO 5, 08028 Barcelona, Spain

Piek Vossen

Department of English, University of Amsterdam
Spuistraat 210, 1012 VT Amsterdam, The Netherlands

Simonetta Montemagni, Elisabetta Marinai

Instituto di Linguistica Computazionale Via della Faggiola 32, I-56100, Pisa, Italy

January 1992

ESPRIT BRA-3030 ACQUILEX Working Paper No. 043

Multilingual lexical representation

Ann Copestake, Bernie Jones, Antonio Sanfilippo
Computer Laboratory, University of Cambridge
New Museums Site, Pembroke Street, Cambridge, CB2 3QG, UK
Ann.Copestake@cl.cam.ac.uk, bemj1@phx.cam.ac.uk,
Antonio.Sanfilippo@cl.cam.ac.uk

Horacio Rodríguez
Dept. Llenguatges i Sistemes Informatics, Universitat Politècnica de Cataluyna
PAU GARGALLO 5, 08028 Barcelona, Spain

Piek Vossen
Department of English, University of Amsterdam
Spuistraat 210, 1012 VT Amsterdam, The Netherlands

Simonetta Montemagni, Elisabetta Marinai
Instituto di Linguistica Computazionale Via della Faggiola 32, I-56100, Pisa, Italy

Abstract

We describe an approach to the representation of lexical translation equivalence which allows cross-linguistic generalisations to be expressed within a typed unification-based formalism, and illustrate how a large scale multilingual lexical knowledge base that uses this representation may be constructed semi-automatically from machine readable dictionaries.

Topic areas: the lexicon, machine translation.

1 Introduction

This paper describes the multilingual aspects of the lexical knowledge base (LKB) system which has been designed and implemented as part of the ACQUILEX project[1] to allow the representation of syntactic and semantic information extracted from machine readable dictionaries (MRDs). The LKB's representation language (LRL) is based on typed feature structures; Copestake (1992) and Sanfilippo and Poznanski (1992) describe the LKB and the LRL and illustrate how large scale monolingual lexicons have been constructed semi-automatically in the LKB for subsets of the four languages involved (English, Spanish, Dutch and Italian), using a common type system. In this paper we describe our use of the LRL to represent multilingual information in the form of links between translation equivalent lexical entries, and discuss the techniques we have implemented to allow large numbers of such links to be generated (semi-)automatically from MRDs.

Recently, there has been considerable interest in encoding multilingual transfer

[1] 'The Acquisition of lexical knowledge for Natural Language Processing systems' (ESPRIT BRA-3030); see Briscoe(1991).

rules in unification-based formalisms, aiming for declarativeness and bidirectionality, but allowing sufficient expressiveness to deal with lexical 'gaps'; specialisation, and so forth (e.g. Zajac, 1989; Estival *et al.*, 1990; Alshawi *et al.*, 1991). Our concept of translation equivalence maintains these advantages, but abstracts away from MT transfer rules; it is intended to allow the expression of linguistic and lexicographic generalisations in constructing a multilingual lexicon which would support a variety of approaches to MT and could also be used by human translators, language teachers, lexicographers and linguists, given suitable interfaces. MT transfer rules do not necessarily explicitly link the monolingual lexical entries; however this is essential for the multilingual LKB to be useful more generally.

The mechanism for expressing equivalence between feature structures described here is quite general and abstract but our use of it has similarities with the 'Shake-and-Bake' approach to MT (Whitelock, 1991; Beaven, 1992) because of its strongly lexicalist orientation and its reliance on unification of semantic indices. However, in addition, our formalism makes it possible to state generalisations about translation equivalence between classes of lexical entries and cross linguistic relationships between lexical processes. Our long term goal is to develop a linguistically-motivated theory of the multilingual lexicon, capturing the appropriate generalisations and lexical rules. Sanfilippo *et al.* (1992) explore an account of MT transfer similar to that of Beaven and Whitelock using a multilingual lexicon constructed in the framework presented below.

We define lexical translation equivalence in terms of cross-linguistic links, *tlinks*, between the lexical entries in the monolingual lexicons. Although in the simplest and commonest cases unmodified pairs of lexical entries can be treated as translation equivalents, in general we have to allow for 'mismatches' such as differences in argument ordering, plurality, and specificity of reference, and for 'lexical gaps', where a word sense in one language has to be translated by a phrase in the other. We use inheritance from both lexical entries and rules in tlinks; this makes them compact while ensuring that the multilingual and monolingual components are compatible. The tlink mechanism does not involve any extensions to the LRL.

In the next section we briefly introduce the salient features of the LRL. We then describe the tlink mechanism and discuss some examples of the representation of translation equivalence between nouns. In section 4 we illustrate how we can generate at least the simple links between the monolingual entries (semi-)automatically by making use of bilingual MRDs and conclude by briefly discussing the use of the LKB in MT.

2 The Lexical Representation Language

The LRL is fully described in papers in Briscoe et al (1992, forthcoming) and various aspects are discussed in Briscoe (1991), Copestake and Briscoe (1991), Copestake (1992) and Sanfilippo and Poznanski (1992). Here we therefore only give a brief informal description of the relevant aspects.

The LRL is based on typed feature structures similar to those described by Carpenter (1990, 1991). However we adopt a slightly different notion of well-formedness. In our system every type must have exactly one associated feature structure (FS) which acts as a *constraint* on all FSs of that type; all well-formed FSs of that type must be subsumed by the constraint. The constraint also defines which features are *appropriate* for a particular type; a well formed FS may only contain appropriate features. Constraints are inherited by all subtypes of a type, but a subtype may introduce new features (which will be inherited as appropriate features by all its subtypes). A constraint on a type is a well-formed FS of that type; all constraints must therefore be mutually consistent.

An example of a monolingual lexical entry, derived automatically from LDOCE (Longman Dictionary of Contemporary English, Procter (1978)) entry for chocolate (in the drinking chocolate sense) is shown below (Figure 1).

```
chocolate_L_1_4
<sense-id : dictionary> = "LDOCE"
<sense-id : homonym-no> = "1"
<sense-id : sense-no> = "4"
<sense-id : ldb-entry-no> = "5902"
<cat : m-feats : agr : pers> = 3
<cat : m-feats : count> = BOOLEAN
<cat : m-feats : agr : num> = SG
<lex-noun-sign rqs> < DRINK_L_2_1 <lex-noun-sign rqs> .
```

Figure 1

(Bold font is used for types, small capitals for features, the boxes indicate parts of the FS which are not shown completely. In the path notation < is used for default inheritance from a named FS.) The type system and default inheritance mechanism allows a relatively compact lexical entry to be expanded into a detailed FS. The syntactic and formal semantic portions of the lexical sign are relatively conventional, however we also encode detailed lexical semantic information, the RQS ("relativised qualia structure", Calzolari 1991). All the monolingual lexicons in the LKB make use of the same type system.

We also encode grammar and lexical rules as typed feature structures, which represent relationships between two or more signs (i.e. FS descriptions of single words or phrases). A lexical rule is a FS of type **lexical-rule** which is a subtype of **rule** (**top** is the most general type).

```
rule (top)
< id > = rule-id
< 0 > = sign
< 1 > = sign.

lexical_rule (rule)
< 0 > = lex_sign
< 1 > = lex_sign.
```

The expanded constraints for these types are:

$$\begin{bmatrix} \textbf{rule} \\ \textsc{rule-id} = \textbf{rule-id} \\ 0 = \textbf{sign} \\ 1 = \textbf{sign} \end{bmatrix} \qquad \begin{bmatrix} \textbf{lexical\_rule} \\ \textsc{rule-id} = \textbf{rule-id} \\ 0 = \textbf{lex\_sign} \\ 1 = \textbf{lex\_sign} \end{bmatrix}$$

Thus all lexical rules have to have the features 0 and 1 which must both have values which are of type **lex_sign**. Lexical rules can be regarded as a means of generating new lexical signs; if a lexical entry can be unified with the FS at the end of the path `<1>` in the lexical rule, then the FS at the end of the path `<0>` is a new lexical sign. Alternatively they can be regarded statically, as expressing the relationship between two existing lexical signs. We use lexical rules both to represent morphological derivation and sense extension (Copestake and Briscoe 1991, Briscoe and Copestake 1991).

3 Representation of translation equivalence

We represent cross-linguistic relationships between word senses as tlinks. In general there may be a many-to-many equivalence between word senses, but each possibility is represented by a single tlink. In the common, simple cases, word senses may be represented as being directly translation equivalent, and it is unnecessary to augment the monolingual information, other than simply to assert that a link is present (see **simple-tlink**, below). The tlink mechanism, however, allows the monolingual information to be augmented with translation specific information, in a variety of ways.

A tlink is simply a FS of type **tlink**, which is to be read as stating that two FSs (the "output structures") are to be regarded as translation equivalents. The tlink encodes the relationship between the input word senses and these output structures; it can be viewed as describing how lexical entries may be transformed into translation equivalent pairs. A complete tlink is essentially a relationship between two **rule**s (as defined above) where the rule inputs have been instantiated by the representations of the word senses in the source and target languages which are to be linked and where the rule outputs are translation equivalent. The type **tlink** is defined as follows:

```
tlink (top)
< tlink-id > = tlink-id
< sfs > = rule
< tfs > = rule
< sfs : 0 : sem : ind > = < tfs : 0 : sem : ind >.
```

The third line indicates equivalence of the variables in the two output structures in the particular monolingual encoding of semantic information that we are currently adopting. For all tlinks the FSs at the end of the paths `< sfs : 0 >` and `< tfs : 0 >` will be translation equivalent. For all tlinks at least the paths `< sfs : 1 >` and `< tfs : 1 >` have to be instantiated by lexical entries to produce the complete tlink[2]

By defining types of tlinks the concept of translation equivalence can be constrained and generalisations can be encoded. The commonest and simplest cases of translation equivalence can be represented as **simple-tlink**s.

```
simple-tlink (tlink)
< sfs : 0 > = < sfs : 1 >
< tfs : 0 > = < tfs : 1 >.
```

A **simple-tlink** is applicable in the case where two lexical entries which denote single place predicates (nouns etc) are straightforwardly translation equivalent, without any transformation being necessary. (For verbs more argument equivalence specifications are necessary, see Sanfilippo *et al.* (1992).)

Like lexical rules tlinks can be regarded statically or dynamically as generating new FSs; given a FS in one language, and an appropriate tlink, unification with the FS at the end of the appropriate path (eg `< sfs : 0 >`) in the tlink, will result in the FS at the end of the other output path being returned (eg `< tfs : 0 >`).

Assuming that the LDOCE sense *chocolate* 1 4, is translation equivalent to the Van Dale *chocolade* 0 2, we would have the tlink:

```
simple-tlink
< sfs : 1 > << chocolate_L_1_4
< tfs : 1 > << chocolade_V_0_2.
```

[2]Tlinks and lexical rules are both symmetrical and reversible; we use the terminology source (sfs), target (tfs), input (1) and output (0) solely for ease of exposition.

where <<, indicates non-default inheritance from a named FS (the particular lexical entries). The "syntactically sugared" version, which appears in files, is:

```
chocolate_L_1_4 / chocolade_V_0_2
simple-tlink .
```

Some restrictions on translation can be expressed by making the target or source FSs more specific. For example both *maestro* and *maestra* in Spanish can be translated as *teacher* in English; the restriction that *maestro* denotes a male teacher and *maestra* a female one, can be encoded as follows:

```
teacher_1 / maestro_1 :
simple-tlink
< sfs : 0 : rqs : sex > = male.

teacher_1 / maestra_1 :
simple-tlink
< sfs : 0 : rqs : sex > = female.
```

Alternatively we can define a type **human-tlink** and state as a constraint that the values for the SEX feature must be equivalent in the translation equivalent feature structures.

```
human-tlink (simple-tlink)
< sfs : 0 : rqs : sex > = < tfs : 0 : rqs : sex >.
```

The restrictions would then follow, assuming that the Spanish lexical entries were appropriately instantiated, and would apply to the whole class.

Somewhat rarer and more complex cases of linking arise when the changes to the FS are those such as pluralisation, which is a process that has to be represented separately, and which has to be viewed as a transformation of a FS rather then a simple restriction. For example, a lexical/morphological rule for plural formation, which would be formulated anyway for use in parsing/generation, can be used in a tlink: we encode the idea that the equivalence is to be defined between a basic lexical entry and a lexical entry after rule application, by instantiating one half of the tlink with the appropriate lexical rule.

For example *furniture* can be encoded as translation equivalent to the plural *muebles* by specifying that the named rule 'plural' has to be applied to the base sense in Spanish.

```
furniture_1 / mueble_1 :
tlink
< sfs : 0 > = < sfs : 1 >
< tfs > << plural <> .
```

This tlink can be represented figuratively as shown in Figure 2; unlabelled arrows indicate token identity between FSs.

$$
\begin{tikzcd}
\text{SFS1} \arrow[r] & \text{SFS0} \arrow[r, leftrightarrow, "\textbf{tlink}"] & \text{TFS0} & \text{TFS1} \arrow[l, "\text{plural}"'] \\
\text{furniture} & \text{furniture} & \text{muebles} & \text{mueble}
\end{tikzcd}
$$

Figure 2

Since the singular form of *mueble* would not unify with the feature structure at the end of the output path `< tfs : 0 >` a translation of "mueble" as "furniture" would not be generated by this tlink. The translation of "mueble" as "a piece of furniture", for example, would be encoded as a distinct tlink.

In general when a word has to be translated by means of a phrase, a representation for that phrase can be built up in the tlink, to make an entire phrase translation equivalent to a word sense. This mechanism is used in the case of 'lexical gaps'. For example Spanish *novillo* can be translated as *young bull*; the translation can be represented as follows (where 'forward-application' is a grammar-rule through which adjectives and nouns are combined):

```
young_1 bull_1 / novillo_1 :
tlink
< sfs > << forward-application <>.
```

(The use of a specific phrase in a tlink does not commit an MT system to generate exactly that phrase; the meaning representation could be used by a system that generated from logical form, for example.)

In some cases the existence of a tlink between two lexical items implies a further translation relationship. For example, in English, there is a regular sense extension such that a word used primarily to denote an animal can also be used to denote the meat of that animal (e.g. *lamb*, *chicken*, see Copestake and Briscoe (1991)). A similar sense extension rule applies to Italian (Östling 1991) but in Dutch a compound is generally used (*lam*, *lamvlees*). To represent the relationship between these lexical rules we define the type **tlink-rule**:

```
tlink-rule (top)
< id > = tlink-rule-id
< t0 > = tlink
< t1 > = tlink
< srule > = lexical-rule
<srule : 1 > = < t0 : sfs : 1 >
<srule : 0 > = < t1 : sfs : 1 >
< trule > = lexical-rule
<trule : 1 > = < t0 : tfs : 1 >
<trule : 0 > = < t1 : tfs : 1 >
```

By stating that the lexical rule for animal-grinding is linked with that for compounding with *vlees*, we can for example automatically generate the relationship between `lamb_2` and `lamvlees` from a simple tlink between `lamb_1` and `lam`, see Figure 3. [3]

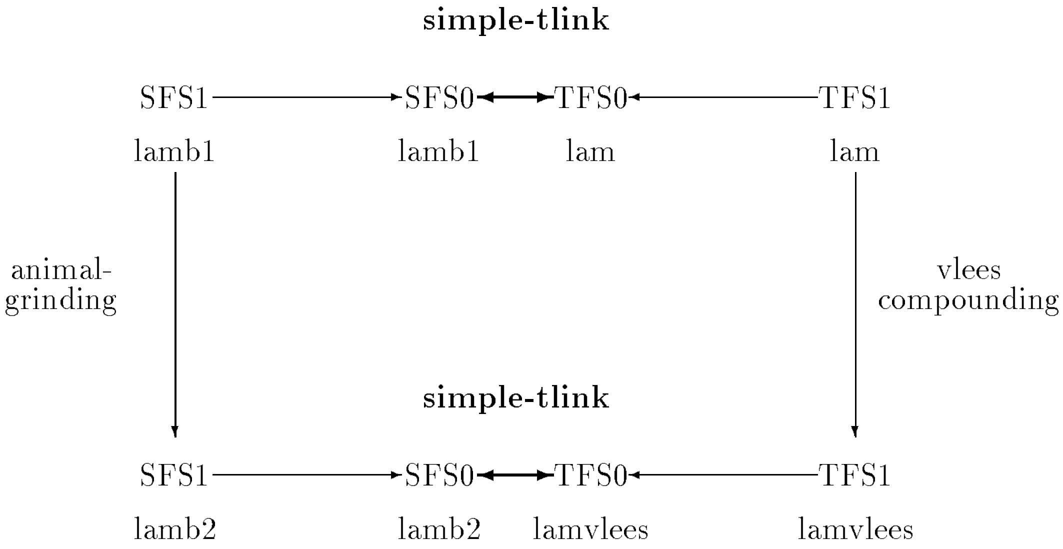

Figure 3

There are many examples of such correspondences; for example the English sense extension between trees and their fruits (*pear* etc) is mirrored in Italian with a gender distinction; the trees are masculine but the fruits feminine (*pero*, *pera*)[4].

Our earlier examples can in fact be described more generally. A tlink-rule can be defined for human-denoting nouns so that given the equivalence between two entries of one gender the tlink for the other gender can be produced. Since gender in English is normally unmarked, one instantiation of this is with a 'null lexical rule', which states equality between two FSs. Thus given the tlink between *teacher* and *maestra* we can automatically generate that between *teacher* and *maestro*, see Figure 4.

[3] Since we are just making use of the monolingual sense extension mechanism here we can rely on that to handle cases where the sense extension is blocked (e.g. *pig*).

[4] It does not necessarily matter for translation purposes whether the rule can fully predict the effects of the sense extension; even if the rule is used statically to encode the regular aspects of the relationship between two lexicalised items, an appropriate translation link will be generated if the monolingual processes are sufficiently similar.

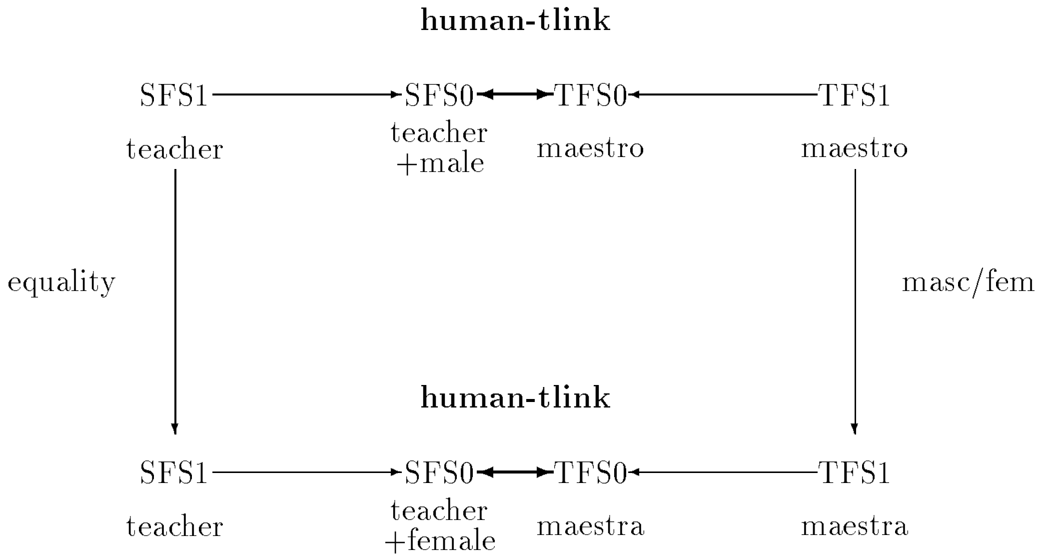

Figure 4

The count-mass discrepancy which is responsible for the need to invoke pluralisation in the furniture-muebles example can also be generalised. Rather than manually specifying the translations of both the singular and plural count nouns for every case we can specify that the translation equivalent of the singular noun will be some more complex construction which individuates the mass noun, equivalent to the use of a phrase such as "piece of", see Figure 5.

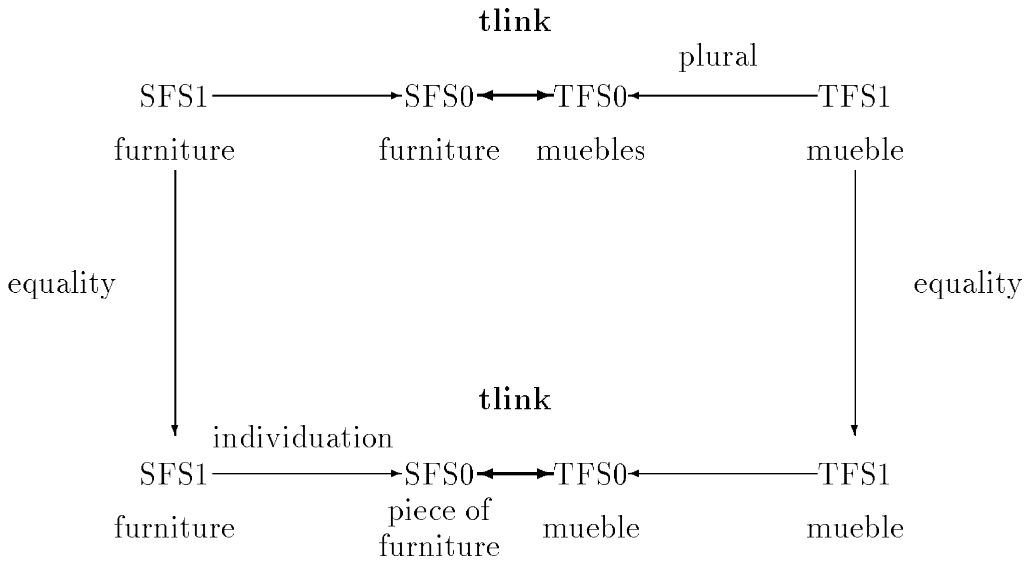

Figure 5

4 Finding translation equivalents

The basic difficulty with automatic generation of tlinks (or any representation of lexical translation equivalence) from MRDs is selection of the correct sense. We are attempting to link entries derived from monolingual dictionaries. Bilingual dictionaries, where they do discriminate between senses, may use a scheme which differs from any monolingual, and the translations given in bilinguals typically have no sense marking. Consider the following, taken from the English—Spanish VOX dictionary (Biblograf 1988):

> **crush** *s.* compresión, presión, aplastiamiento, machacadura, estrujamiento, estrujón. *2* cantidad de material machacado, estrujado, etc...

Not all of these translations will be appropriate for all the senses given in LDOCE for *crush*, and even those which are appropriate may also have inappropriate senses.

We thus use bilingual dictionaries to give a range of possible LKB entries in a target language, when given an LKB entry in the source language. Currently this is done using orthographic information alone and we can only generate simple tlinks automatically. This candidate list of translations is input to a matching program, LUCIFER, which returns the sense or senses most semantically similar to the source sense.

4.1 Lucifer

LUCIFER is a general purpose utility for comparison of typed FSs (see Copestake and Jones (1991) for a fuller description). A statistic is assigned to each candidate FS, with the magnitude of the statistic proportional to the quality of match. This is compiled by descending both FSs simultaneously, comparing structure along all the branches. If in the course of the FS descent any mismatches are found, these will decrease the statistic for the match. On reaching the terminal nodes, a numeric value is returned dependent on the similarity of the values at equivalent leaf nodes on the two structures. The values are defined as follows:

Exact match	+1.0
Source subsumes target	+0.5
Target subsumes source	+0.5
Type mismatch	-Total

Thus subsumption is taken to indicate a similarity between the FSs, but less so than an exact match between the types.. A type incompatability at any node gives a negative value of the magnitude of the statistic of the whole FS, were it to match perfectly. This ensures that FSs containing a serious mismatch will always have a small statistic compared with those that do not contain such a mismatch.

The problem with assigning a simple statistic is that no distinction is made in the importance of the components of a FS. Orthography, for example, should be ignored in cross-linguistic comparison of lexical entries. Thus each branch of the matched FSs has a unique weight, ranging between 0.0 and 1.0, which modifies the values returned. Weights are multiplied by the nodal value of their descendant node to give a set of inputs to the parent node. These are summed to form the nodal value of the non-terminal parent node. The process is then repeated in successive ancestral levels until the root node of the FS is assigned a nodal value. This value is used as the statistic qualifying the whole match.

Weights cannot be assigned by hand, as we are dealing with large numbers of entries and comparison of different classes of words may involve different specific features being highlighted as important. Hence a learning procedure is used to modify the weights for the specific matching task being carried out and this involves training, either by human confirmation of LUCIFER's initial choices or by using examples of unique translations from the bilingual.

All weights are initialised with a value of 0.5 and updated on the basis of the correct matching FS(s). If there are any mismatches in the structure, the weight of the branches leading to this mismatch is decreased - if there is a match the weight is increased. As the mismatch is far more serious than the match the weight is decreased by far more than than it is increased in the case of a match. The formula used is a variation on the error signal for the logistic activation function in classic feed-forward back-propagation networks using the generalised delta rule (Rumelhart et al. 1986). If the weight on a branch is ω_i

$$\partial\omega_\tau = \eta\frac{\alpha_\tau}{Total}\omega_\tau(1-\omega_\tau)$$

where α_i is the activation of the descendant node and η is the learning rate.

4.2 Generating tlinks

We have implemented a system which combines LUCIFER with an interface to the bilingual dictionaries. After the initial training phase matching will only involve human intervention when LUCIFER cannot find clear winners because no FS has a sufficiently high score. The full paper will report the results of several large scale experiments to generate simple tlinks automatically for the subsets of the vocabulary for which we have constructed monolingual lexicons.

In many cases the various candidates given by the bilinguals are distinguishable on the basis of their lexical semantic type alone since we have adopted a relatively fine-grained classification; thus the full power of LUCIFER's matching algorithm is not needed. For example the senses of *crush* given in LDOCE divide into the incompatible lexical semantic types, **dur_event** (non-point like event), **c_subst_nat** (comestible natural substance) and **abstract** (we have not yet attempted any subclassification of abstract, non-event denoting, nouns).

However there are cases where more detailed comparison of the FSs is necessary; *chocolate* in English can refer to the solid substance or to drinking chocolate and both these senses share the same rqs type **c_art_subst**, whilst *cioccolato* in Italian is specific to the solid substance, *cioccolata* is used for the substance or the drink, and *cioccolatino* for the individual sweet. From the bilingual we get possible links from chocolate_L_1_4 to all the senses of these three words. *cioccolatino* has rqs type **c_art_obj** which is incompatible with that of chocolate_L_1_4 and it is therefore given a very low matching statistic. LUCIFER discriminates between the other possibilities on the basis of the different value for PHYSICAL-STATE and for the telic role. Thus the translation link will be made between chocolate_L_1_4 and cioccolata_G_0_2 (see Figure 1 and Figure 6).

5 Machine translation

If the output forms in the tlink alone are considered they are essentially equivalent to transfer rules between FSs. Existing MT systems could make use of the tlinks by transforming these parts of the tlink into transfer rules in the format required. (The issues of defining correspondences between features used in the LKB and those of other systems are essentially the same as in the monolingual case, and so are not considered further here.) For example the ELU system (Estival et al 1990), uses lexical transfer rules which could be derived from tlinks on the assumption that transfer variables are equivalent to reentrancy between tlink output FSs. The BCI system (Alshawi et al 1991) uses transfer between quasi logical forms; the appropriate lexical transfer rules between predicates can be derived from tlinks under the same assumptions. Since at its most general the tlink mechanism states correspondences between FSs its use is not specific to transfer based MT. In our current LKB the predicates in the logical forms are not only language specific but refer to senses in particular MRDs. Eventually we would hope to avoid this, but a 'deep' transfer/interlingua approach, based on Wilks(1991) suggestion that LDOCE senses be used in an interlingua, might still be able to utilise an LKB built using our current techniques.

As mentioned in the introduction Whitelock and Beaven's 'Shake-and-Bake' approach is the most natural way of utilising the LKB in MT. The LRL allows improvements on their approach in that generalisations can be expressed using the type system. Sanfilippo *et al.* (1992) explore this in more detail, and describe approaches to translation mismatches involving language-specific lexicalisation patterns for verbs using the mechanisms presented here. However we regard MT as only one possible application; ultimately the multilingual LKB should be able to support the needs of linguists, lexicographers and others who need access to substantial quantities of cross-linguistic information and who wish to express cross-linguistic generalisations about the lexicon.

References

Alshawi, H., D. Carter, M. Rayner and B. Gambäck (1991) 'Translation by Quasi Logical Form Transfer', *Proceedings of the 29th ACL,* Berkeley, California, pp. 161–168.

Beaven, J. L. (1992) *Lexicalist Unification-Based Machine Translation,* PhD thesis, University of Edinburgh.

Biblograf (publisher) (1988) *VOX Diccionario Inglés-Español Español-Inglés,* Briscoe, T. (1991) 'Lexical Issues in Natural language Processing' in E. Klein and F. Veltman (eds.), *Natural Language and Speech,* Springer-Verlag, pp. 39–68.

Briscoe, T. and A. Copestake (1991) 'Sense extensions as lexical rules' in D. Fass, E. Hinkelman and J. Martin (eds.), *Computational approaches to non-literal language: metaphor, metonymy, idiom, speech acts, implicature,* Proc. of IJCAI Workshop, Sydney, pp. 12–20.

Briscoe, T., A. Copestake and V. de Paiva (eds.) (1992, forthcoming) *Default Inheritance in Unification-Based Approaches to the Lexicon,* CUP.

Calzolari, N. (1991) 'Acquiring and representing semantic information in a lexical knowledge base', *Proceedings of the ACL SIGLEX Workshop on Lexical Semantics and Knowledge Representation,* Berkeley, California, pp. 188–197.

Carpenter, B. (1990) 'Typed feature structures: Inheritance, (In)equality and Extensionality', *Proceedings of the First International Workshop on Inheritance in Natural Language Processing,* Tilburg, The Netherlands, pp. 9–18.

Carpenter, R. (1991, in press) *The Logic of Typed Feature Structures,* Cambridge University Press, Tracts in Theoretical Computer Science.

Copestake, A. (1992) 'The ACQUILEX LKB: representation issues in semi-automatic acquisition of large lexicons', to appear in *Proceedings of the ANLP-92,* Trento, Italy

Copestake, A. and T. Briscoe (1991) 'Lexical Operations in a Unification Based Framework', *Proceedings of the ACL SIGLEX Workshop on Lexical Semantics and Knowledge Representation,* Berkeley, California, pp. 88-101.

Copestake, A. and B. Jones (1991) *Support for multi-lingual lexicons in the LKB system,* ms University of Cambridge, Computer Laboratory.

Estival, D., A. Ballim, G. Russell and S. Warwick (1990) 'A syntax and semantics for feature structure transfer', *Proceedings of the 3rd International Conference on theoretical and methodological issues in MT of NLs,* Austin,Texas, pp. 131–143.

Östling, A. (1991) *Sense extensions in the Italian food subset,* ACQUILEX working paper.

Procter, P. (ed) (1978) *Longman Dictionary of Contemporary English,* Longman, London.

Rumelhart, D. E., G. E. Hinton and R. J. Williams (1986) 'Learning Internal Representations by Error Propagation' in D. E. Rumelhart and J. L. McClelland (eds.), *Parallel Distributed Processing — Explorations in the microstructure of Cognition Volume 1,* Cambridge, Massachusetts, pp. 318–362.

Sanfilippo, A., T. Briscoe, A. Copestake, T. Marti and A. Alonge (1992) *Translation equivalence and lexicalisation,* ms University of Cambridge, Computer Laboratory.

Sanfilippo, A. and V. Poznanski (1992) 'The Acquisition of Lexical Knowledge from Combined Machine-Readable Dictionary Sources', to appear in *Proceedings of the ANLP-92,* Trento, Italy

Whitelock, P. (1991) *Shake-and-bake translation,* submitted to Coling-92.

Wilks, Y. (1991) *More introduction and a bibliography,* Machine Translation: ACL tutorial notes.

Zajac, R. (1989) 'A transfer model using a typed feature structure rewriting system with inheritance', *Proceedings of the 27th ACL,* Vancouver, BC, pp. 1–7.