MatsLex - a Multilingual Lexical Database for Machine Translation

Jörg Tiedemann

Department of Linguistics
Uppsala University
Box 527
S-75 120 Uppsala
Sweden
joerg@stp.ling.uu.se

Abstract

MatsLex represents a relational database which can be used to store multilingual lexical data in a central and coherent lexicon. Tools and interfaces have been implemented to maintain the database and to apply its contents to different multilingual applications. MatsLex has been developed to feed different modules of a machine translation system with appropriate data, monolingual as well as bilingual. The database gives the user full control of the lexicon. In the paper, features and interfaces of the database are discussed as well as the connection to the machine translation engine.

1. Overview

MatsLex is a multilingual lexical database that has been developed in the MATS project at Uppsala University/Sweden (MATS, 2000; Sågvall Hein et al., forthcoming; Weijnitz, forthcoming). The primary aim of the project is the scaling up of the transfer-based machine translation prototype MULTRA (Beskow, 1993; Sågvall Hein, 1997) for one domain. For this purpose, lexical resources have been derived from corpora (Tiedemann, 1999) and stored in the MatsLex database. The database is designed to provide a flexible and coherent environment for storing and managing multilingual lexical data, and for linking them bilingually. The internal structure of the lexicon is based on a relational database model. The database can be queried and updated via transparent database views in web-based interfaces.

MatsLex is the central store of all the lexical data available for the translation process, and from this "runtime lexicons" such as bilingual link lexicons are compiled. For consistency, modifications are allowed in the central database only whereas runtime lexicons are strictly read-only. An overview of the MatsLex database and its connection to the machine translation system is sketched in figure 1.
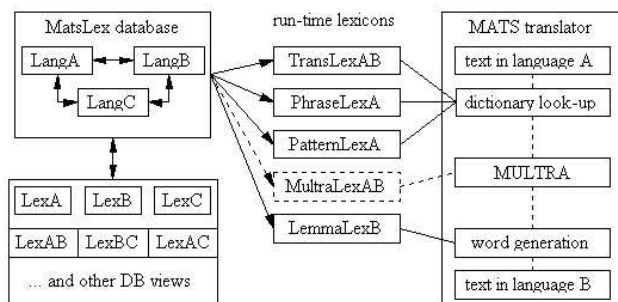

Figure 1: MatsLex and MULTRA.

The structure of the database will be presented in section 2., database views and their functionality are presented in section 3.1.. The compilation of run-time lexicons as in-put for the machine translation system is then presented in section 3.2..

2. The Database Structure

The lexical database comprises a set of tables with morphological, syntactic, and semantic information with appropriate relations between them. The relational structure of a monolingual part of the database is shown in figure 2.
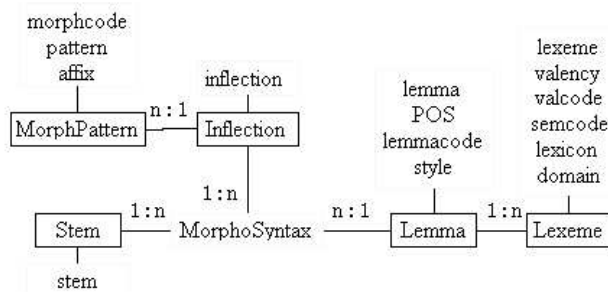

Figure 2: The Monolingual database structure.

Morphosyntactic and semantic information is commonly expressed by feature structures. In the MATS database, compressed, compositional codes are used as short-cuts for feature structures. Codes are defined for expressing morphosyntactic features (morphcode), lemma-specific features (lemmacode), semantic features (semcode), and valency related features (valcode).

Surface wordforms are not included explicitly in the database. MatsLex stores inflectional patterns instead and surface words are derived from these patterns and their technical stems. The crucial point of this approach is to define accurate paradigms and to correctly link lexical entries to appropriate paradigms. Generalised patterns may not be suitable for all languages but in the worst (but most unlikely) case each entry would have its own paradigm. The advantage of this approach is to make updating the database easier. All the possible surface forms are included implicitly when a lemma enters the database and is linked to a

paradigm. The morphological paradigms in MatsLex are labelled by representative words and their inflectional patterns are defined in the database table 'MorphPattern' by regular-expressions. The 'pattern' field specifies a regular expression to be matched against the technical stem and the 'affix' field holds the modification to be made in the creation of the wordform. In many cases (in Swedish and English) this simply means concatenating appropriate suffixes with the technical stem (consider table 1). Other languages such as German need more complex modifications, e.g. the German word 'weggefahren' with the technical stem 'weg+fahr' can be created by adding the suffix 'en' to the end and substituting '+' with the prefix 'ge'. Even phrasal surface forms such as 'fahre weg' can be created in the same paradigm by adding the suffix 'e', substituting '+' with a single space and reordering both parts.

stem	pattern	affix	surface word
install	`$`	`"s"`	installs
install	`$`	`"ing"`	installing
weg+fahr	`^(.*)\+(.*)$`	`"$1\ge$2\en"`	weggefahren
weg+fahr	`^(.*)\+(.*)$`	`"$2\e $1"`	fahre weg

Table 1: Inflectional patterns with regular expressions.

Another distinctive feature of the database is the possibility to use regular expressions as technical stems that match classes of similar tokens with the same morphosyntactic and semantic features. Constructions with a general pattern are, e.g., dates, time-expressions, and numbers. Some examples are given in table 2

stem	examples
`([0-9]+),?([0-9]*)(\%)`	50,5% ; 99%
`([0-9]*1):a`	1:a; 261:a
`([0-9]+):e`	9:e; 764:e
`([0-9]{2})\/([0-9]{2})\/([0-9]{2})`	01/03/04

Table 2: Token classes defined by regular expressions.

The MatsLex database stores each table from the monolingual lexicon with a language prefix. Hereby, data for additional languages can be added easily to the database. The advantage of keeping several languages in parallel in one central database is the possibility of linking them together. To accomplish this, MatsLex allows the establishment of bilingual links between lexemes from different languages. The structure of such links within the relational framework is shown in figure 3.
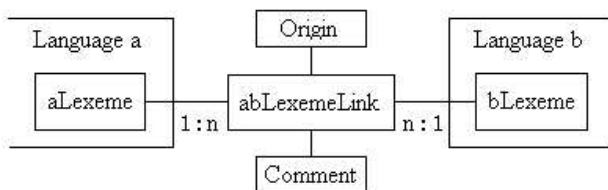

Figure 3: Bilingual links between lexemes.

Origin and comment are basically used for adding the

possibility of giving some extra information for each link. Both are optional and may give the origin of the established link (user name, link tool, etc.) and a descriptive comment. This is especially useful for manual validation of automatically added links.

Regular expressions are stored as strings in the same way as ordinary stems in the database. They are marked with a special style code together with the associated lemma. They can be linked to syntactic and semantic information as other stems and their associated lexemes can be linked to corresponding items in other languages as well. However, regular expression stems represent sets of items and therefore there has to be a special function which translates possible items into the correct correspondences in other languages. This is implemented in the form of substitutions of matched strings.

Swedish

stem	`([0-9]{2})\-([0-9]{2})\-([0-9]{2})`
word	99-12-01

American English

lemma	`$2\/$3\/$1.xx`
word	12/01/99

German

lemma	`$3\.$2\.$1.xx`
word	01.12.99

Table 3: Translations of regular expressions.

An example is given in table 3. The regular expression which is given as the Swedish stem matches a class of Swedish date expressions. The substitutions are given as lemmas in other languages[1]. Consider the Swedish example date which is given in the table: '99-12-01'. The date expression is matched by the regular expression and the substitutions which are given for American English and German change the format into the language specific formats (appropriate punctuation and order).

3. Database Access

3.1. Updating the Database

The MatsLex database can be queried and updated via a web interface. A library of database access functions (RDBstream) has been implemented which allows transparent access to different database views. Views are defined as subsets of table columns within the database. The relations between tables are defined in specific configuration files. The RDBstream library uses the internal database structure as specified in the configuration files in order to run appropriate SQL commands according to the task and the current view to the database. RDBstream includes common database access functions such as 'select' (for querying the view), 'insert' (for adding a data record), 'delete' (for removing data records), and 'update' (for modifying data records). The access library creates necessary joins between tables and solves possible inconsistencies in case

---

[1] The prefix which follows the last dot in the lemma specifies the part-of-speech of the lemma. The prefix '.xx' corresponds to an undefined POS.

of database modifications according to the database view and the relational database structure. This way, it is easy to add new database views to the web interface. Any subset of table columns can define a database view with all the access possibilities such as 'select', 'insert', 'delete', and 'update'. However, updating a database view is not always possible. Certain views may not allow updates because they may cause inconsistencies in the database. In such cases, the consistency check will fail and the update will be dismissed.
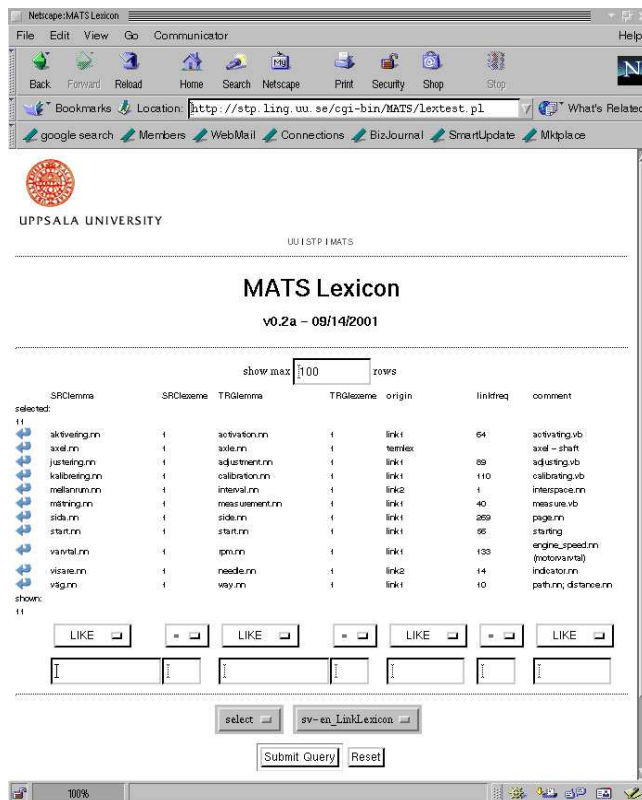

Figure 4: The MatsLex Web Interface.

The current MatsLex database contains data for English and Swedish and links between lexemes of both languages. The web interface includes five views for each language and one view for lexeme links between the two languages. A new language pair can easily be integrated by running a setup script which creates appropriate tables in the relational database and which also creates similar views for the new language(s) and the bilingual links. New data can be added and accessed immediately in the same way as existing data in the database.

3.2. Compiling Run-time Lexicons

The MatsLex database is the central storage space of both monolingual and bilingual lexical data. Modifications are done within this framework only in order to provide a consistent data collection. Several subsets of the data in the database are used in the MATS machine translation engine (MATS translator)(Weijnitz, forthcoming). Lexical data are taken directly from the database and are stored in so-called run-time lexicons (see figure 1). Compiling such lexicons has several advantages. First of all, data are stored in exactly the format which is required by certain components of the translator. Secondly, run-time lexicons contain only necessary information and do not have to handle other data. Both facts enable a very fast access to the data which speeds up the translation process. Another advantage is consistency of the data. Run-time lexicons are compiled at a certain time from the then current data collection. They are strictly read-only and are independent from the central database. Modifications in the database do not have immediate influence on the behaviour of the translator. All changes are introduced first after another compilation of the run-time lexicons.

Compiling run-time lexicons is done by specific scripts which use the 'select' function of the RDBstream library. This library makes it very easy to modify the subset of data which is needed for certain purposes. It also makes the access to the database transparent, i.e. changes of the internal database structure basically do not cause changes of the compiled lexicons as long as all necessary information is included in the database. For example, if all information would be stored in one joined database table only, then the compilation script could still be used as before without any modifications.

Currently, two compilation scripts have been implemented which produce four run-time lexicons that are used by the MATS translator: GenWordDB and GenLemmaDB.

The GenWordDB script creates three lexicons which are used by the dictionary look-up function of the MATS translator. One run-time lexicon (TransLexAB) contains all words and phrases in language A and their morphosyntactic description and their translations into language B (if any are found in the database). Another lexicon (PhraseLexA) contains all initial words of possible phrases with pointers to the actual phrase constructions. This is used to speed up the detection of phrasal expressions. The third lexicon (PatternLexAB) contains morphosyntactic descriptions of regular expressions (as in table 2) and their translation patterns if there are any.

The GenLemmaDB script compiles a lexicon (LemmaLexB) of all lemmas and their morphosyntactic codes with pointers to the corresponding surface wordforms. This lexicon is used for generating the actual sentence from the output of MULTRA (see figure 1).

A third script is yet to be implemented for compiling a run-time translation lexicon (MultraLexAB) which can be used for transfer and generation within MULTRA. MULTRA applies transfer rules which may cause lexical changes. For this purpose, MULTRA requires access to a translation lexicon in order to pick additional items which are needed for the translation. Currently, this lexicon is stored in the former MULTRA formalism. Internal changes in MULTRA are needed in order to enable the engine to use a run-time lexicon similar to the one which is used in the dictionary look-up module.

4. Application

The MatsLex database is applied in an on-going project at Uppsala University, KOMA (KOMA, 2001). Currently, it is used to store domain specific data for Swedish and English. In particular, data have been taken from a corpus of

Swedish and English truck maintenance documents which have been provided by the truck and bus manufacturer Scania CV AB in Södertälje/Sweden. Words from this corpus have been analysed morphosyntactically with the Uppsala Chart Parser (UCP) (Sågvall Hein, 1983) and linked automatically to their English counterparts (Löfling, 2001). All this material has been used to scale up the Swedish/English translation lexicon by adding translations to the previously created Swedish lexicon (Almqvist and Sågvall Hein, 1996; Almqvist and Sågvall Hein, 2000). Links in the lexicon have been validated by professional translatorsand missing links have been added manually (Forsbom, forthcoming b). The linked items in the translation lexicon were the starting point for the English lexicon. A system of morphological paradigms has been developed for English and has been added to the database (Forsbom, forthcoming a).

The old MULTRA demonstration lexicon of 369 Swedish and 184 English lemmas has been extended to 20,883 Swedish lemmas and 6,562 English lemmas. Furthermore, the number of Swedish wordforms has been increased from 36,827 as collected from the corpus to 123,212 due to the generation of word forms using inflection patterns (Karlsson and Thaning, 2001). The number of English wordforms which are represented in the current database amounts to 26,241. The database is growing, new items are added frequently. Additional syntactic and semantic information will be added in the on-going project.

5. Conclusions

MatsLex represents a relational lexical database for multiple languages. Its main purpose is to store multilingual data which is used by the MATS translation system. It is used as the central and coherent database for collecting and maintaining lexical data. Tools and interfaces have been developed to search and update the database. Database views and web interfaces provide a convenient environment for accessing the lexicon and for consistent modifications. The database gives the user full control of the lexicon which is used in the translation system. The data in the lexicon serve several tasks in the translation system such as monolingual and bilingual dictionary queries and wordform generation from morphosyntactic descriptions. For this purpose, run-time lexicons are generated from the database which provide efficient access to necessary data.

The current database contains domain-specific items in English and Swedish. They have been extracted from a corpus of the corresponding domain. Links between Swedish and English have been established automatically and validated professionally. All items in the database have been morphosyntactically analysed and associated with appropriate inflectional paradigms. The database content is growing and additional syntactic and semantic information is added gradually.

The database also makes use of patterns in the form of regular expressions. They can be used to represent classes of similar tokens such as numerical expressions and dates. Such regular expressions are powerful tools which improve the coverage of the lexicon significantly without including all items explicitly.

MatsLex has been developed for a specific task, namely machine translation, and it has been filled with domain-specific items. However, the system is open to new applications and other languages can be integrated easily.

7. References

Ingrid Almqvist and Anna Sågvall Hein. 1996. Defining ScaniaSwedish – a controlled language for truck maintenance. In *Proceedings of the 1st International Workshop on Controlled Language Applications (CLAW'96)*, pages 159–164, Centre for Computational Linguistics, Katholieke Universiteit, Leuven.

Ingrid Almqvist and Anna Sågvall Hein. 2000. A language checker of controlled language and its integration in a documentation and translation workflow. In *Proceedings of the Twenty-Second International Conference on Translating and the Computer*, Translating and the Computer 22, Aslib/IMI, London, November 16–17.

Björn Beskow. 1993. Unification-based transfer in machine translation. Reports from the Department of Linguistics, RUUL #24, Uppsala University.

Eva Forsbom. forthcoming-a. Scaling up the generation lexicon for Multra. Project report.

Eva Forsbom. forthcoming-b. Scaling up the transfer lexicon for Multra. Project report.

Stina Karlsson and Sten Thaning. 2001. Automatisk generering av ordformer till Scania-databasen [Automatic generation of wordforms for the Scania database]. Project report.

KOMA. 2001. The KOMA project. http://www.ida.liu.se/~nlplab/koma/.

Camilla Löfling. 2001. Att skapa ett lemmalexikon för manuell och maskinell översättning [To create a lemma lexicon for manual and automatic translation]. Master's thesis, Uppsala University.

MATS. 2000. The MATS project. http://stp.ling.uu.se/mats.

Anna Sågvall Hein, Eva Forsbom, Jörg Tiedemann, Per Weijnitz, Ingrid Almqvist, Leif-Jöran Olsson, and Sten Thaniing. forthcoming. Scaling up an MT prototype for industrial use - databases and data flow.

Anna Sågvall Hein. 1983. A parser for Swedish. status report for SVE.UCP, February. UCDL-R-83-1, Center for Computational Linguisics, Uppsala University.

Anna Sågvall Hein. 1997. Language control and machine translation. In *Proceedings of the 7th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-97)*, Santa Fe, USA.

Jörg Tiedemann. 1999. Word alignment - step by step. In *Proceedings of the 12th Nordic Conference on Computational Linguistics*, Trondheim/Norway.

Per Weijnitz. forthcoming. MATS - a platform for machine translation. Project report.