Stephan Bopp, Institut für Informatik, Universität Basel & Lexicologie, Vrije Universiteit Amsterdam

Phrase Manager: a System for the Construction and the Use of Multi-word Unit Databases

Abstract

Phrase Manager (PM) is a system for the specification and the use of databases of multi-word units. Phrase Manager covers the mapping between text words and dictionary entries where there is no one-to-one relation between these entities. PM is an extension of Word Manager, a system that handles inflectional and derivational morphology. PM allows the specification of classes of multi-word units and the construction of dictionaries of phrasal expressions by relating individual units to these classes. This paper will briefly discuss the mapping process and, then, focus on the formalism for the specification of multi-word unit classes and multi-word unit entries.

1. Word Analysis in NLP-Systems

The processing of words in NLP-systems can be summarised as follows: The basic entity is the lexeme (according to the definition in Matthews (1974), where it is a citation form with its inflectional paradigm). It occupies a central position between two independent mapping processes:

- Mapping 1 between strings as they occur in texts and lexemes
- Mapping 2 between lexemes and different readings, i.e. distinct information a lexeme is associated with in the dictionary of a particular NLP-system.

Mappings 1 and 2 are independent from each other. Since system-specific and theory-dependent information is concentrated in mapping 2, mapping 1 is much more independent of a particular application of an NLP-system. Therefore, a system – like the one presented here – that covers mapping 1 can be used (re-used!) by a variety of NLP-systems that, then, "only" have to specify the system-specific information for mapping 2.

In the mapping from text words to lexemes, we distinguish the following cases:

- Inflection: A text word is directly analysed as a wordform of the inflectional paradigm of a lexeme.
- Word formation: A text word is analysed as a wordform of the inflectional paradigm of a lexeme formed from existing lexemes by the application of word-formation rules.
- Clitics: A text word is analysed as (orthographic) contraction of two or more lexical words and split up for further analysis.
- Multi-Word Units: Two or more text words are analysed as elements of one lexical unit and combined in a phrasal expression.

In our system, Word Manager (WM) covers the first two cases: inflection and word formation of single text words. The following sections will concentrate on Phrase Manager (PM), the part of the system that covers the last two types of mapping. A detailed technical description of WM and PM is given in Domenig & ten Hacken (1992) and Pedrazzini (1994) respectively.

2. Development and Use of a WM/PM-Database

WM and PM are at the same time a database system and a knowledge specification environment. Starting from a new, empty database, the development of a WM/PM-database is divided into the following two stages:

- Creation of a rule database: The linguist's interface provides a formalism for the specification of morphological and phraseological rule knowledge. Each rule specification contains at least one example entry.
- Creation of a morphological dictionary database: The lexicographer's interface supports the specification of entries to the database (entry knowledge). Each entry is assigned to a rule specified by the linguist.

The two interfaces are dedicated user interfaces supporting the specification tasks of the linguist and the lexicographer. In practice, the

two specification stages affect each other in that the lexicographer's work may call for modifications in the original rule database where the latter turns out to be inadequate or incomplete. The specifications of the two types of knowledge remain clearly separated, however, because different interfaces to the database are used.

WM/PM follows a client-server model: The databases are maintained on a server. The access to the databases is handled by "tailor-made" interfaces that are designed to provide the exact type and form of morphological and phraseological information a particular client application needs.

3. Analysis in a PM-Database

PM is subdivided into three modules called *Clitics*, *PerInfl* and *Idioms*. The typical interaction between these modules and WM is shown in Fig. 1.

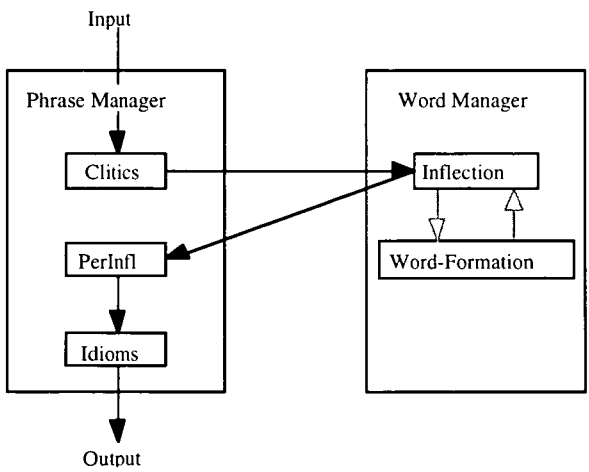

Fig. 1: Analysis in a WM/PM-Database

The input is a sentence consisting of text words. The *Clitics* module identifies text words that consist of more than one lexical word and splits them up. Most words will not be affected, but examples like English *cannot*, Italian *andarci* ('to go there') or German *vom* ('of the') will be replaced by *can not*, *ci andare* and *von dem*. WM analyses the resulting list of words morphologically. The result is a list of wordforms with, for each wordform, the indication of the lexeme it belongs to and its relevant

grammatical features (number, tense, etc.). When there is more than one correct analysis, all are passed on.

PerInfl analyses two or more wordforms as one wordform of a single-word lexeme and adds the relevant grammatical features. It recognises e.g. *has broken* as the 3rd person singular perfect tense of the verb *break*. We call this *Periphrastic Inflection*.

Finally, *Idioms* identifies possible multi-word expressions. It will propose an idiomatic analysis for the sequence *has broken the ice* (*to break the ice*) together with the "literal" analysis of the single words. The final output of PM consists of a list of lemmatised and morphologically analysed wordforms and of proposals for analytic wordforms and phrasal expressions.

4. Knowledge Specification in PM

Before the lexicographer can specify entries to a phrasal dictionary of a particular language, the linguist has to formulate rules for clitics and periphrastic inflections as well as a set of phrasal expression classes. This section will concentrate on the formalism that assists the linguist in this task.

4.1 Clitics

The *Clitics* module splits up text words into two or more lexical words. Avoiding the controversial discussion on the exact definition of clitics, we pragmatically assume that every text word that consists of more than one lexical word contains at least one element that could be defined as a clitic in the broadest sense of the word. In PM, *clitic* is an orthographic term.

For the specification of cliticised words, the linguist has to define so-called CElements (Clitic Elements) and Clitic Rules. The Clitic Rules define how CElements forming a text word are to be replaced by CElements that correspond to lexical wordforms. The example is a simplified representation of the specifications for Italian *dello* = *di lo* ('of the'). First, the CElements are specified:

```
(CElement Prep.di.citation)   "di" (Cat Prep)
(CElement Prep.di.variant)    "de"
(CElement Art.lo.citation)    "lo" (Cat Art-Def)(Num Sing)(Gend M)
(CElement Art.lo.variant)     "llo"
```

The CRule responsible for the analysis and the replacement of *dello* by *di* and *lo* is then specified as follows:

(CElement Prep.di.variant) + (CElement Art.lo.variant)
= (CElement Prep.di.citation) , (CElement Art.lo.citation)

The first line specifies the structure of the complex text word. The orthographic concatenation is indicated by "+". The second line defines the wordforms that will replace the text word. The comma indicates that they are written as two separate words. When the string *dello* is analysed, this Clitic Rule will replace it by *di* and *lo*. All strings are passed on for further analysis, but WM will not recognise *dello* as an independent wordform and therefore rule it out. In other cases, when there is ambiguity, both the separated strings and the unmodified text word are presented as results of the analysis. This is the case with Italian *colla* that can either be the noun *colla* ('glue') or the contraction of the preposition *con* ('with') and the feminine singular article *la*.

4.2 Periphrastic Inflection

The *PerInfl* module is responsible for the analysis of what traditional grammars usually call analytic wordforms. PM enables the linguist to specify rules that 1) combine two (or more) text words, 2) attribute this lexical wordform to the paradigm of a lexeme and 3) attach the relevant grammatical features. The (simplified) example rule is responsible for the analysis of the comparative form of Italian adjectives (e.g. *più bello* 'more beautiful'):

(Cat Adv) {"più"}, (Cat Adj) = (Pos 2)(CForm 2)(Perc 2)(Degree Comp)

Any sequence of the adverb *più* and an adjective will be combined. This periphrastic wordform has the same position (Pos) in the sentence and the same citation form (CForm) as the second element, i.e. the adjective. The wordform features of the combined wordform consist of the percolated features (Perc) of the adjective and the explicitly mentioned feature (Degree Comp):

"più" (Cat Adv), "bello" (Cat Adj)(Num SG)(Gender M)
= "più bello" (Cat Adj)(Num SG)(Gender M)(Degree Comp)

Again, both the complex wordform and the two single wordforms will be passed on for further analysis.

4.3 Idioms

The *Idioms* module handles the specification and analysis of phrasal expressions of any kind (henceforth *idioms*). The linguist can specify so-called idiom classes, i.e. classes of idioms sharing a particular phrase structure. The syntactic structure is the primary criterion for the structuring of these classes. Secondary criteria are modification in word order and internal modification.

PM does not provide a parser for syntactic structures. Therefore, the linguist has to specify the syntactic structure of the different idiom classes, however, only as deep as it is relevant for the syntactic modifications the idioms can undergo. E.g.:

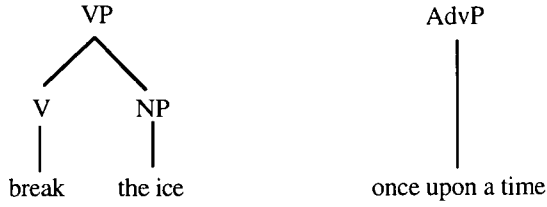

Fig. 2: Syntactic structures in PM

Word order can be altered in the first example (e.g. in *the ice has been broken*), whereas it is invariable in the second example.

Let's have a look at a possible specification of the idiom class *break the ice* belongs to:

```
(PHClass VP.V+NP)
  SYNTAX TREE
  (VP (V NP))
  MODIFICATIONS
  V >
  TRANSFORMATIONS
  Inversion
```

The rule name (PHClass VP.V+NP) identifies the class within the database. The specification of the class itself is subdivided into three main parts:

- *Syntax Tree*: Specification of the relevant syntactic structure (see above) of the idioms belonging to the class.
- *Modifications*: Definition of discontinuation, i.e. the possibilities of adding "external words" without damaging the idiomatic character of the expression. Here, the verb can be modified by external words on its right-hand side.
- *Transformations*: Note that *transformations* does not refer to the same term used in some versions of transformational grammar. In PM, transformations simply handle modifications in word order like e.g. inversion in passive constructions (*the ice has been broken*), etc. The actual specification is done elsewhere in the rule database, with the labels of the syntax tree definition:

  Inversion
  (VP (V NP)) -> (VP (NP V))

A fourth part is the example definition. For testing and consistency reasons each class specification obligatorily contains at least one example phrase. In fact, the examples belong to the entry knowledge specification (cf. section 2). Therefore they can also be seen as an illustration for internal modifications that can be specified with each phrasal dictionary entry. Our particular example would be specified as follows:

EXAMPLE
<break> the ice

The pointed brackets indicate that *break* may be inflected. Here, all wordforms of the paradigm are acceptable. For other cases, additional restrictions can be formulated, like e.g. when passivization is not allowed:

EXAMPLE
<kick> the bucket
kick(Cat V)^(Voice Passive)

where "^" stands for not. Other restrictions in example phrases are indicated by round brackets (variants) and square brackets (optional words):

EXAMPLE
(<sell> <go>) like hot cakes

EXAMPLE
<mind> ONE'S [own] business

ONE'S in the second expression is an example of another specification tool: key-words. Key-words are associated with feature sets that identify groups of (wordforms of) lexemes. Here, the keyword would obviously be associated with all possessive determiners.

4.4 Specification of PM-Entries

Once the knowledge specification described in the previous sections is accomplished, idioms can be added to build up a phrasal dictionary. When the lexicographer decides that a group of words constitutes an idiom, he/she chooses an idiom class and specifies an entry definition in the manner described for the example entries above.

5. Output of a PM-Analysis

The output of a PM-analysis of the sentence *The ice has finally been broken* illustrates what kind of information one can expect to find in a PM-database. PM offers different views on the analysed data:

- The sentence after the application of the clitic rules and the periphrastic inflection analysis:

  the ice has_been_broken finally
  he has_arrived not_yet (input sentence: *he hasn't arrived yet*)

- The wordforms with their citation form and their morphological features according to the specifications in WM and PM:

the	the(Cat Art)
ice	ice(Cat N)(Num Sing)
has been broken	break(Cat V)(Voice Passive)(Tense Perf)(VForm 3rd-Sing)
finally	finally (Cat Adv)

- When a wordform is the result of the application of Clitic Rules or PerInfl-Rules, the indication of the rules:

  has been broken AUXILIARY: have pp
  PASSIVE: be pp

Additionally, when the sentence contains an idiom stored in the PM-database:

- The minimal form ("citation form") of the idiom:
  break the ice
- The idiom class it belongs to:
  (PHClass VP.V+NP)
- The syntax tree of the idiom as it appears in the analysed sentence (according to the syntax definition in the corresponding idiom class) and the indication of Transformations that have been applied (if any):

```
Syntax-Tree: (VP
                    (NP the ice)
                    (V has been broken)
Transformation:     INVERSION
```

These are but a few examples of how the information stored in a WM/ PM-database can be accessed and represented. The database structure and the flexible, highly theory-independent knowledge specification allow other views and representations according to the needs of different human users and NLP-systems.

6. Conclusion

During the last years, WM and PM have been developed, fully implemented and tested in several prototyping cycles. Comprehensive WM rule-databases have been implemented for Italian, French, German and English (Bopp 1993, Bopp 1994, Brunner 1991, Gregorio 1993, Gupta 1989). The implementations served as a basis for PM-specifications of Italian, German and English. These test implementations contain from a couple of idiom classes (English) up to several hundred classes (German). The tests proved that the PM-formalism is powerful enough to cover a wide range of different types of idioms, to build up multi-word unit dictionaries and to analyse cliticised words, periphrastic inflection clusters and phrasal expressions. We therefore believe to have developed a powerful tool for the handling of word processing beyond the limits of simple text words.

The WM/PM system has been implemented in CLOS (Common Lisp Object System), and runs on Macintosh computers. The server application has also been ported to Sun SPARCstations. A demonstration version is available through FTP, at ftp.ifi.unizh.ch, in the directory /pub/mac/WordManager.

References

Bopp, S. (1993): *Computerimplementation der italienischen Flexions- und Wortbildungsmorphologie*, Olms Verlag, Hildesheim.

Bopp, S. (1994): "An Implementation of Italian Inflection and Word Formation", in *Proceedings Euralex-94*, Amsterdam, 30/8–3/9.

Brunner, C. (1991): *An Implementation of English Morphology Using the Program Word Manager*, Lizentiatsarbeit an der Philosophischen Fakultät I der Universität Zürich.

Domenig, M., ten Hacken P. (1992): *Word Manager: A System for Morphological Dictionaries*, Olms Verlag, Hildesheim.

Gregorio, S. (1993): *Implementation of English Inflectional and Derivational Morphology*, Lizentiatsarbeit am Institut für Informatik der Universität Basel.

Gupta, A. (1989): *La formalisation de la morphologie française sur la base du système Word Manager*, Lizentiatsarbeit an der Philosophischen Fakultät I der Universität Zürich.

Matthews, Peter H. (1974): *Morphology: An Introduction to the Theory of Word Structure*, Cambridge University Press, Cambridge.

Pedrazzini, Sandro (1994), *Phrase Manager: A System for Phrasal and Idiomatic Dictionaries*, Olms Verlag, Hildesheim.