Chapter 12
Identification and Lexical Representation of Multiword Expressions

Jan Odijk

12.1 Introduction

Multi-word Expressions (MWEs) are word combinations with linguistic properties that cannot be predicted from the properties of the individual words or the way they have been combined. MWEs occur frequently and are usually highly domain-dependent. A proper treatment of MWEs is essential for the success of NLP-systems. This will be the topic of Sect. 12.2.

Generic NLP-systems usually perform less well on texts from specific domains. One of the reasons for this is clear: each domain uses its own vocabulary, and it uses generally occurring words with a highly specific meaning or in a domain-specific manner. For this reason, state-of-the-art NLP systems usually work best if they are adapted to a specific domain. It is therefore highly desirable to have technology that allows one to adapt an NLP system to a specific domain for MWEs, e.g., on the basis of a text corpus. Technology is needed that can identify MWEs in a maximally automated manner. This will be discussed in Sect. 12.3.

An NLP-system can only use an MWE if it is represented in a way suitable for that NLP system. Unfortunately, each NLP system requires its own formats and properties, and the ways MWEs are represented differs widely from NLP-system to NLP-system. Therefore a representation of MWEs that is as theory- and implementation-independent as possible and from which representations specific to a particular NLP system can be derived in a maximally automated manner is highly desirable. A specific approach to this, based on the Equivalence Class Method (ECM) approach, and applied to Dutch, will be described in Sect. 12.4.

J. Odijk (✉)
UiL-OTS, Trans 10, 3512 JK Utrecht, The Netherlands
e-mail: j.odijk@uu.nl

P. Spyns and J. Odijk (eds.), *Essential Speech and Language Technology for Dutch*,
Theory and Applications of Natural Language Processing,
DOI 10.1007/978-3-642-30910-6_12, © The Author(s) 2013

Using the method for the automatic identification of MWEs and the method for lexically representing MWEs, a database of MWEs of the Dutch language called *DuELME* has been constructed. It will be described in Sect. 12.5.

We end with concluding remarks in Sect. 12.6.

12.2 Multiword Expressions

Multi-word Expressions (MWEs) are word combinations with linguistic properties that cannot be predicted from the properties of the individual words or the way they have been combined. A word combination can, for example, have an unpredictable meaning (*de boeken neerleggen*, lit. 'to put down the books', meaning 'to declare oneself bankrupt'), it can have only limited usage (e.g. *met vriendelijke groet* 'kind regards', used as the closing of a letter), or it can have an unpredictable translation (*dikke darm* lit. 'thick intestine', 'large intestine'), etc.

MWEs do not necessarily consist of words that are adjacent, and the words making up an MWE need not always occur in the same order. This can be illustrated with the Dutch MWE *de boeken neerleggen* 'to declare oneself bankrupt'. This expression allows a canonical order with contiguous elements (as in (1a)), but it also allows other words to intervene between its components (as in (1b)), it allows permutations of its component words (as in (1c)), and combinations of permutations and intervention by other words that are not components of the MWE (as in (1d)):

(1) a. Saab heeft gisteren **de boeken neergelegd**
lit. 'Saab has yesterday the books down-laid'

b. Ik dacht dat Saab gisteren **de boeken** wilde **neerleggen**
lit. 'I thought that Saab yesterday the books wanted down-lay'

c. Saab **legde de boeken neer**
lit. 'Saab laid the books down'

d. Saab **legde** gisteren **de boeken neer**
lit. 'Saab laid yesterday the books down'

In addition, certain MWEs allow for (and require) controlled variation in lexical item choice, e.g. in expressions containing bound anaphora such as *zijn geduld verliezen* 'to lose one's temper', where the possessive pronoun varies depending on the subject (cf. *Ik verloor mijn/*jouw geduld, jij verloor *mijn/jouw geduld*, etc.), exactly as the English expression *to lose one's temper*.

Of course, not every MWE allows all of these options, and not all permutations of the components of an MWE are well-formed (e.g. one cannot have **Saab heeft neergelegd boeken de* lit.'Saab has down-laid books the').

One can account for such properties of MWEs by assigning an MWE the syntactic structure that it would have as a literal expression: it will then participate in the syntax as a normal expression, and permutations, intrusions by other words or phrases, etc. can occur just as they can occur with these expressions under their literal interpretation.[1] Adopting this approach for the Dutch MWE *de boeken neerleggen* accounts immediately for the facts in (1) and for the ill-formedness of the example **Hij heeft neergelegd boeken de* given above, since this latter string is also ill-formed under the literal interpretation.

State-of-the art NLP systems do not deal adequately with expressions that are MWEs, and this forms a major obstacle for the successful application of NLP technologies. Reference [16] is titled: *Multiword expressions: a pain in the neck for NLP* and states that "Multiword expressions are a key problem for the development of large-scale, linguistically sound natural language processing technology".

Three problems must be solved to overcome this. First, an NLP system must have an implemented method of dealing with MWEs. This topic will not be dealt with in this paper. A lot of research has been spent on this, and for the purposes of this paper we simply observe that it has resulted in a wide variety of approaches in different grammatical frameworks and different implementations (see [13] for some relevant references).

Second, an NLP system does not 'know' which combinations of words form MWEs. Just providing a list of MWEs (an MWE lexicon) will not in general suffice because each domain has its own vocabulary and its own MWEs. There must thus be a way of dynamically creating MWE lexicons by identifying MWEs in new text corpora in a maximally automated way. The approach to this problem adopted here will be described in Sect. 12.3.

Third, each MWE identified must be represented lexically. The approach adopted here for this problem will be described in Sect. 12.4, and takes into account the fact that solutions to the first problem come in many different varieties in a wide range of grammatical frameworks and implementations.

12.3 Identification of MWEs and Their Properties

We need a method for identifying MWEs in a text corpus in a maximally automated manner. Since component words of an MWE can be inflected, we want to be able to identify multiple combinations of words as instances of the same MWE even if they contain differences with regard to inflection. Since MWEs can consist of nonadjacent words, and since the order in which the components of an MWE occur can vary, we want to be able to identify multiple combinations of words as instances of the same MWE even if the component words are nonadjacent or occur in different orders. Both of these requirements can be met if each sentence of the text corpus is assigned a syntactic structure and each occurrence of an inflected word form is assigned a lemma.

[1] There are, however, additional constraints on MWEs that do not hold for literal expressions. These will either follow from properties of the grammatical system if they involve general restrictions (e.g. they may follow from the fact that components of MWEs often have no independent meaning) or have to be stipulated individually for each MWE with idiosyncratic restrictions.

For experimenting with the identification methods we have used the Dutch CLEF corpus, a collection of newspaper articles from 1994 and 1995 taken from the Dutch daily newspapers *Algemeen Dagblad* and *NRC Handelsblad*. This corpus contains 80 million words and 4 million sentences. We have used the Alpino Parser[2] to automatically annotate each sentence of the corpus with a syntactic structure and each inflected word form token with a lemma.

The identification method takes as input a set of syntactic patterns and a fully parsed text corpus, and outputs a set of candidate MWEs in the form of tuples of lemmas together with a range of grammatical and statistical properties.

For example, for the syntactic pattern *NP_V* it will return tuples consisting of a word of syntactic category *verb* and a word of syntactic category *noun* that are likely candidates to form an MWE with the verb as the head and the noun as the head of the direct object noun phrase, together with statistics on occurring determiners and adjectives modifying the noun, etc.

Based on experiments with various machine learning techniques, it has been decided to apply a binary decision tree classifier to distinguish MWEs from non-MWEs [21].[3] The classifier characterises expressions as an MWE or as a non-MWE using a range of features that reflect or approximate properties of MWEs as reported in the linguistic literature. These include features of lexical affinity between MWE components, local context, morphosyntactic flexibility, and semantic compositionality. Lexical affinity between MWE components has been determined using *salience*, a variant of pointwise mutual information [10], and by a binary feature marking a small set of verbs as *support verbs* [8]. For *local context*, two measures proposed by Merlo and Leybold [12] to quantify *head dependence* are used, viz. the number of verbs that select a given complement, and the entropy of the distribution among the verbs that select for a given complement (cf. [21] for details). In addition, the relative frequency of the label most frequently assigned by the Alpino parser to the dependency relation between the head and the dependent is used. Since in Dutch PP complements are generally closer to the verb in verb-final context than PP adjuncts,[4] the relative frequency of the PP occurring adjacent to the verb group has also been taken into account.

Inflectional modifiability is quantified as follows, following [22]: the most characteristic realisation is simply the realisation of a phrase that occurs most frequently. The degree of modifiability is then expressed as the relative frequency of the most frequent realisation: a low relative frequency for the most frequent realisation indicates high modifiability, a high relative frequency indicates low modifiability.

Another feature used to determine morphosyntactic flexibility is the *passivisation* feature, which simply specifies the relative frequency of the occurrence of the

[2] See http://www.let.rug.nl/vannoord/alp/Alpino/ and [19].

[3] The classifier used is weka.classifiers.trees.j48 [23] which implements the C4.5 decision tree learning algorithm.

[4] [2, p. 107].

candidate expression as a passive. And finally, the pronominalisation feature records whether an NP has been realised as a pronoun.

For semantic compositionality two scores have been used as features, derived from work by Van De Cruys [17], who applied unsupervised clustering techniques to cluster nouns and verbs in semantically related classes. One score (semantic uniqueness) can be characterised as the ratio between the selectional preference of a selector for a selectee and the selectional preference of a selectee for its selector. The second score can be characterised as the selectional association between a selector and a selectee. See [18] for more details.

The data used for testing consist of the Dutch CLEF Corpus and the Twente News Corpus (TwNC[5]), which consists of 500 million word occurrences in newspaper and television news reports and which also has been fully parsed with the Alpino parser. These data have been annotated automatically using the existing lexical databases *Van Dale Lexical Information System* (VLIS)[6] and RBN [11]. The VLIS database contains more than 61,000 MWEs of various kinds (idioms, collocations, proverbs, etc.). From the RBN, app. 3,800 MWEs were extracted and used in the experiments. All expressions from VLIS and RBN have been parsed with the Alpino parser. From the resulting parse structures, sequences of tuples containing word form, lemma, PoS-label and position in the structure have been derived. In the test corpus, all expressions matching the input syntactic pattern have been identified. An absolute frequency threshold has been used to avoid noise introduced by very low frequency expressions. This threshold has to be determined empirically for each pattern as a function of the performance of the classifier.[7] For example, for the pattern *PP_V*, counting only types with frequency ≥10, the test corpus contains 4,969 types and for the pattern *NP_PP_V* it contains 3,519 types. Together this makes 8,488 types covering 1,140,800 tokens. If a candidate expression from the corpus matches with an entry from the set of tuple lists derived from the VLIS and RBN databases, it was marked as MWE, otherwise as non-MWE to serve as the gold standard.[8] Table 12.1 shows the distribution of MWEs and non-MWEs for two patterns.

As one can observe, when one uses a low frequency cut-off (≥10), the proportion of non-MWEs equals 3/4 of the data, while with a higher frequency cut-off (≥50), MWEs and non-MWEs occur equally often (as can be seen for the *NP_V* data in Table 12.1).

Experiments were carried out with different combinations of features. Semantic scores were initially left out. As a baseline, we use a classifier that always selects the

[5] http://wwwhome.cs.utwente.nl/~druid/TwNC/TwNC-main.html

[6] These data have been made available for this research by the Van Dale dictionary publishers. These data could only be used for research internal to the project.

[7] However, it is well known that many individual MWEs occur with very low frequency in a corpus. In fact, [21, p. 18] observes that this may double the number of MWEs. MWE identification methods should therefore also work for low frequency data. We leave this to further research.

[8] No distinction was made or could be made between the literal and the idiomatic uses of an expression. Therefore each expression that can be used as an MWE has been annotated as an MWE.

Table 12.1 Distribution of MWEs and non-MWEs for the patterns *(NP)_PP_V* and *NP_V*

Pattern	Freq	Types	MWEs	non-MWEs
(NP)_PP_V	≥10	8,488	1,910 (22.50 %)	6,578 (77.49 %)
NP_V	≥10	10,211	2,771 (27.13 %)	7,440 (72.86 %)
NP_V	≥50	1,769	917 (51.83 %)	852 (48.16 %)

Table 12.2 Major results for MWE identification for the pattern *(NP)_PP_V*

			MWE			Non-MWEs		
Features	Dataset	Acc	P	R	F	P	R	F
All	Test	82.07	0.62	0.47	0.54	0.86	0.91	0.89
All	All (10fcv)	82.99	0.67	0.48	0.56	0.86	0.93	0.89
All + semantic scores	Test	82.75	0.64	0.49	0.56	0.86	0.92	0.89
All + semantic scores	All (10fcv)	83.40	0.66	0.53	0.59	0.87	0.92	0.89
Baseline	All	77.49	0.00	0.00	0.00	1.00	1.00	1.00

most frequent class. The results specify *accuracy* (Acc), and for each class *precision* (P), *recall* (R) and *F-score* (F). It turned out that using *all* features yielded better results than using any of the tested subsets of features. With semantic scores added, accuracy increased a little more. Evaluation was carried out in two ways: In one set-up 60 % of the data was used for training and 40 % for testing. In a second set-up, all data were used for training and testing using ten-fold cross validation (10fcv). Table 12.2 lists the major results.[9]

The identification method operates on fully parsed sentences. It therefore can use sophisticated grammatical properties as features for the automatic identification of MWEs. The identification method yields a set of tuples that are characterised as MWEs, but it can also provide sophisticated grammatical and statistical properties for each MWE. This has in fact been done, using the CLEF corpus as well as the TwNC. For each candidate expression, a range of properties has been extracted that differs slightly with each pattern, but includes inter alia:[10]

1. The subcategorisation frame assigned by the Alpino parser to the head of the expression;
2. The absolute frequency of the tuple;
3. Corpus size
4. A list of heads of co-occurring subjects with frequencies;
5. For each complement:
    (a) Inflectional properties of the head of the complement, and their frequencies;
    (b) Diminutive information for the head of a nominal complement, and their frequencies;

[9] For more results and details, including an analysis of the relative contribution of the various features, we refer to [21].

[10] See [5, Appendix A] for a detailed description.

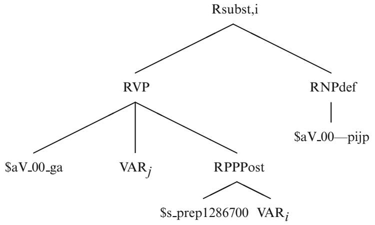

Fig. 12.1 Rosetta D-tree for the idiom *de pijp uitgaan* (simplified)

(c) Determiners co-occurring with the head of a nominal complement, and their frequencies;
(d) Heads of pre-modifiers of the head of the complement, and their frequencies; and
(e) Heads of post-modifiers of the head of the complement, and their frequencies.

The expressions identified as MWEs and their properties (9,451 were identified in the corpora) form the basis for the DuELME lexical database of MWEs, structured in accordance with the ECM.

12.4 Lexical Representation of MWEs

As we have seen above, assigning a syntactic structure to an MWE allows one to account for the fact that MWEs participate in syntax as normal expressions, i.e. allow for permutations, intrusions by other words and phrases, etc. The problem, however, with syntactic structures in NLP systems is that they are highly system specific. This has been shown in detail by Odijk [13] using the Rosetta machine translation system [15] as illustration. The Rosetta system requires, for idiomatic expressions, (1) a reference to a highly specific syntactic structure (cf. Fig. 12.1 for an example), and (2) a sequence of references to lexical entries of the lexicon of the system. In this sequence the presence/absence of these references, the order in the sequence, and the references themselves are all particular to the Rosetta system.

Lexical representations of MWEs that are highly specific to particular grammatical frameworks or concrete implementations are undesirable, since it requires effort in making such representations for each new NLP system again and again and the degree of reusability is low. No *de facto* standard for the lexical representation of MWEs currently exists. Various attempts have been made to develop a standard encoding for certain types of MWEs, especially within the ISLE[11] and XMELLT[12] projects. Reference [13] argues that these attempts are unlikely to be successful, because the structures assigned to the MWEs are highly theory-dependent and even

[11] www.ilc.cnr.it/EAGLES96/isle/

[12] www.cs.vassar.edu/~ide/XMELLT.html

within one grammatical framework, there will be many differences from implementation to implementation. Since most syntactic structures are fully specified tree structures, they are difficult to create and maintain. Reference [3] outlined an approach to represent MWEs in a form which can support precise HPSG, and which is also claimed to be reasonably transparent and reusable. Though their approach may work for certain types of MWEs, they fail to come up with a satisfying solution for representing MWEs.

The central idea behind the ECM is that a standardised representation does not prescribe the structure of an MWE, but backs off to a slightly weaker position, viz. it requires that it is specified which MWEs have the same syntactic structure. In short, it requires that equivalence classes of MWEs are created, based on whether they have the same syntactic structure. Having these equivalence classes reduces the problem of assigning a concrete structure and properties to an MWE to doing this for one instance of the class. And for this problem, the ECM includes a procedure that specifies how to derive that information to a large extent from the concrete system in which the MWE is incorporated.

The ECM thus specifies (1) a way to lexically represent MWEs, and (2) a procedure to incorporate MWEs into a concrete NLP system in a maximally automated manner.

An ECM-compatible lexical representation consists of

1. An MWE pattern, i.e. an identifier that uniquely identifies the structure of the MWE. The equivalence classes are defined with the help of these MWE patterns: MWEs with the same pattern belong to the same equivalence class;
2. A list of MWE components. This takes the form of a sequence of strings, each string representing the lexicon citation form of an MWE component. As to the order of the components, the proposal leaves the order free, but only imposes the requirement that the same order is used for each instance in the same equivalence class;
3. An example sentence that contains the MWE. The structure of the example sentence should be identical for each example sentence within the same equivalence class.

Next to the MWE description, we need a description of the MWE patterns. This is a list of MWE pattern descriptions, where each MWE pattern description consists of two parts:

1. An MWE pattern, and
2. Comments, i.e. free text, in which it is clarified why this MWE pattern is distinguished from others, further indications are given to avoid any possible ambiguities as to the nature of the MWE structure. It is even possible to supply a more or less formalised (partial) syntactic structure here, but the information in this field will be used by human beings and not be interpreted automatically.

This concludes the description of the lexical representation of an MWE in accordance with the ECM. Table 12.3 shows three instances of the same MWE equivalence class from Dutch, and gives a description of the MWE pattern used to define this equivalence class

Table 12.3 Three instances of MWE equivalence class *Pat1* and the description of the equivalence class

Pat.	Components	Example	Gloss	Translation
Pat1	de pijp uit gaan	Hij is de pijp uitgegaan	He is the pipe out-gone	'He died'
Pat1	het schip in gaan	Hij is het schip ingegaan	He is the ship in-gone	'He had bad luck'
Pat1	de boot in gaan	Hij is de boot ingegaan	He is the boat in-gone	'He had bad luck'

Pat.	Description
Pat1	Verb taking a subject and a directional adpositional phrase (PP). This PP is headed by a postposition and has as its complement a noun phrase consisting of a determiner and a singular noun.

The procedure to convert a class of ECM-compatible MWE descriptions into a class of MWE descriptions for a specific NLP-system consists of two parts: a manual part, and an automatic part. The manual part has to be carried out once for each MWE pattern, and requires human expertise of the language, of linguistics, and of the system into which the conversion is to be carried out. The automatic part has to be applied to all instances of each equivalence class.

The manual part of the conversion procedure for a given MWE pattern *P* consists of five steps:

1. Select an example sentence for MWE pattern *P*, and have it parsed by the system; select the right parse if there is more than one;
2. Define a transformation to turn the parse structure into the idiom structure;
3. Use the result of the parse to determine the unique identifiers of the lexical items used in the idiom;
4. Use the structure resulting from the parse to define a transformation to remove and/or reorder lexical items in the idiom component list;
5. Apply this transformation and make sure that the citation form of each lexical item equals the corresponding element on the transformed citation form list.

Observe that only one part of the first step of this procedure (*Select the right parse*) crucially requires human intervention.

The automatic part of the conversion procedure is applied to each instance of the equivalence class defined by idiom pattern *P*, and also consists of five steps:

1. Parse the example sentence of the idiom and check that it is identical to the parse tree for the example sentence used in the manual step, except for the lexical items;
2. Use the transformation defined above to turn the parse tree into the structure of the idiom;
3. Select the unique identifiers of the lexical items' base forms from the parse tree, in order;
4. Apply the idiom component transformation to the idiom component list;
5. Check that the citation form of each lexical item equals the corresponding element on the transformed idiom Component list.

The application of these procedures to real examples has been illustrated in detail in [13, 14]. However, the ECM as described above has one serious drawback, which can be solved by extending it with parameters into the *parameterised ECM*. A concrete example may help illustrate the problem and how the use of parameters resolves it. MWEs can contain nouns. In Dutch, nouns can be singular (*sg*) or plural (*pl*), and positive (*pos*) or diminutive (*dim*). In the ECM as described above a different equivalence class would be needed for each of these four cases (and even more if more than one noun occurs in a single MWE). By introducing two parameters for nouns (*sg/pl*, *pos/dim*), it is possible to group these four equivalence classes into a single equivalence superclass, and to have a single pattern for this superclass, which is parameterised for the properties of the noun (*sg/pl*; *pos/dim*). The extension with parameters introduces a little more theory and implementation specificity to the method, but it does so in a safe way: NLP systems that can make use of these parameters will profit from it, while systems that cannot make use of these parameters are not harmed since the original equivalence classes can still be identified. For the example given above the theory or implementation dependency that is introduced is that properties such as *sg/pl* and *pos/dim* on a noun are dealt with by rules applying to just the noun. It can be expected that many different grammatical frameworks share this assumption. The extension contributes to reducing the number of equivalence classes and increasing the number of members within equivalence classes. It will therefore reduce the number of MWEs that have to be dealt with manually and increase the number of MWEs that can be incorporated into an NLP system in a fully automatic manner. Reference [13] argued that the parameterised ECM is a feasible approach on the basis of data from English and a small set of data from Dutch, and the work reported on here for Dutch has confirmed this.

12.5 The DuELME Lexical Database

An ECM-compatible lexical database for Dutch MWEs has been created [6]. It is ECM-compatible because it classifies MWEs in equivalence classes based on their syntactic structure and uses lexical representations that cover all items required by the ECM. The database is corpus-based: the expressions included have been selected on the basis of their occurrence in the CLEF and TwNC corpora.

Six syntactic patterns frequently occurring in the parsed VLIS database have been selected as input patterns for the MWE identification algorithm. These patterns are defined in terms of dependency triples <head, dependent, dependency label> consisting of a *head* word and a *dependent* word, each of a specific syntactic category, and a *dependency label* to characterise the dependency relation between the words – cf. Table 12.4, where identical subscripts indicate that the same word must be involved and where *compl* is a variable for a range of labels that the dependency relation between a complement PP and a verb can have in Alpino syntactic structures.

Table 12.4 Input patterns

Pattern	Description
NP_V	<verb, noun, direct object>
(NP)_PP_V	<verb$_i$, adposition, *compl*> and optionally <verb$_i$, noun, direct object>
NP_NP_V	<verb$_i$, noun, indirect object> and <verb$_i$,noun, direct object>
A_N	<noun, adjective, modifier>
N_PP	<noun, adposition, modifier>
P_N_P	<adposition, noun$_i$, complement> and <noun$_i$, adposition, modifier>

Table 12.5 Number of candidate expressions and absolute frequency threshold by pattern

Pattern	Threshold	Count
NP_V	$f \geq 10$	3,894
(NP)_PP_V	$f \geq 10$	2,405
NP_NP_V	$f \geq 10$	202
A_N	$f \geq 50$	1,001
N_PP	$f \geq 30$	1,342
P_N_P	$f \geq 50$	607
Total		**9,461**

With these patterns as input, a wide variety of expression types can be extracted, since the patterns are underspecified for a lot of aspects, such as determiners, adjectival and adverbial modifiers, inflectional properties, etc. The resulting set of MWEs therefore requires many more MWE patterns than these six for an adequate description.

Using these patterns, candidate expressions in the form of tuples of head words and their properties have been identified in the corpora, in the way described in Sect. 12.3. The numbers of identified candidate expressions per pattern as well the absolute frequency thresholds used for each pattern are given in Table 12.5.

The 9,461 candidate expressions and their properties form the input to a process of manual selection of the expressions to include in the DuELME database, and of adapting candidate expressions. The criteria used in this manual selection process are criteria that follow the definition of MWE as given in Sect. 12.2: does the word combination have linguistic properties that cannot be predicted from the properties of the individual words or the way they have been combined? This manual step is necessary for many reasons. First, the identification method does not have 100 % accuracy, so the resulting list contains expressions that cannot be considered MWEs in accordance with the definition of MWEs as given in Sect. 12.2. Second, in many cases the expression as identified by the algorithm is incomplete, e.g., a determiner or modifying adjective that is obligatory for the MWE is not identified as part of the MWE by the identification method. The relevant information to further automate this is available (e.g. the properties contain statistics on the co-occurring determiners, modifiers, etc.), but it has not been used by the MWE identification

Table 12.6 MWEs containing *hand* and *hebben* as components

MWE	Gloss	Translation
zijn handen vol hebben aan	his hands full have on	'be fully occupied with'
de hand hebben in	the hand have in	'be responsible for'
de vrije hand hebben	the free hand have	'be free to do whatever one wants'
een gelukkige hand hebben	a lucky hand have	'be lucky'

Table 12.7 Coverage of ECs

Cov.(%)	#MWEs	#ECs	#parameterised ECs
50	2,616	101	10
60	3,139	166	16
70	3,662	272	25
80	4,186	441	38
85	4,447	572	48
90	4,709	785	63
95	4,970	1,046	87
100	5,232	1,308	140

algorithms.[13] In some cases, one output expression actually covers multiple different MWEs. An extreme case is the candidate expression characterised by the head of the direct object noun *hand* 'hand' and the head verb *hebben* 'have'. The examples from the corpus illustrate four different MWEs having these words as components, as illustrated in Table 12.6. Such examples cannot currently be distinguished as different MWEs by the identification method and have to be split manually into different entries.

The selection process resulted in 5,232 MWEs that have been included in the DuELME database. The selected and improved expressions have been analyzed for a classification by syntactic structure (equivalence classes, ECs). The parameterised ECM has been fully elaborated for Dutch. Eight parameters have been distinguished, each with multiple possible values, and in total 26 possible values.[14] The coverage of the ECs and the parameterised MWE classes is represented in Table 12.7.

In this table, we observe several things. The ECM without parameters requires a substantial amount of ECs to obtain a reasonable coverage, e.g. 785 to cover 90 % (or 4,709) of the lexicon entries. The amount of required ECs is a direct indicator for the amount of effort required to incorporate MWEs into an NLP system in accordance with the ECM procedure, and it is clear that without parameters this is too large to be realistically feasible. By the introduction of parameters, however,

[13] This is an area for future research, which is now possible since the relevant data are available.

[14] See [5, p. 36] for a complete overview.

the number of required ECs reduces dramatically, for 90 % coverage from 785 to 63. Of course, additional effort must be spent to deal with the parameters, but if all parameters can be optimally used, this just adds a fixed one-time effort of 26 operations (corresponding to the number of possible parameter values). This shows that the parameterised ECM approach is feasible, and reduces effort for incorporating MWEs into an NLP system considerably, confirming initial results in this direction presented by Odijk [13].

In the DuELME database, templates for syntactic structures have actually been added for each MWE pattern. These templates for syntactic structures are modeled after the syntactic dependency structures used initially in the CGN (Spoken Dutch Corpus, [7]) as well as in the D-Coi,[15] Lassy[16] and SoNaR[17] projects. These dependency structures have thus become a de facto standard for the representation of syntactic structures for Dutch. Adding such syntactic structures is not needed for the parameterised ECM, but they do no harm either. In fact, they can be beneficial for NLP systems that can deal with them. In particular, the first step of the manual part of the ECM incorporation method ('Select the right parse'), which is the only one requiring human intervention, can now become fully automatic, and thereby the whole manual part can become fully automated. In addition, for systems that use closely related syntactic structures, direct mappings can be defined.

Small experiments have been carried out to test incorporating MWEs into NLP systems. The experiments involved the Rosetta system and the Alpino system. For the Rosetta system this remained a paper exercise, since no running system could be made available. For Alpino, the incorporation worked effectively. It has also been tested, in a very small experiment which can at best be suggestive, how a system with incorporated MWEs performs in comparison to the system without these MWEs. This has been tested by measuring the *concept accuracy per sentence*(CA) as used in [20] for the Alpino system with the original Alpino lexicon and the Alpino system with an extended lexicon:

$$CA^i = 1 - \frac{D^i_f}{max(D^i_g, D^i_p)} \tag{12.1}$$

where D^i_p is the number of relations produced by the parser for sentence *i*, D^i_g is the number of relations in the treebank parse for sentence *i*, and D^i_f is the number of incorrect and missing relations produced by the parser for sentence *i*.

The results, summarised in Table 12.8, show that the concept accuracy of sentences containing an MWE increases significantly in a system with an extended lexicon, and the concept accuracy of sentences not containing MWEs does not decrease (in fact, also increases slightly) in a system with an extended lexicon.

[15] http://lands.let.ru.nl/projects/d-coi/

[16] http://www.let.rug.nl/vannoord/Lassy/

[17] http://lands.let.ru.nl/projects/SoNaR/

Table 12.8 Concept accuracy scores

Sample	Lexicon	CA(%)
MWEs	Alpino lexicon	82.85
	Extended lexicon	94.09
Non-MWEs	Alpino lexicon	95.83
	Extended lexicon	96.39

Such results are encouraging, but because of the small scale of the experiment, it should be confirmed by larger scale experiments before definitive conclusions can be drawn.

The DuELME database, a graphical user interface, and extensive documentation is available via the Dutch HLT Agency.[18] The database has been positively externally validated by CST, Copenhagen, i.e. been subjected to a check on formal and content aspects. In a CLARIN-NL[19] project, the database has been stored in a newly-developed XML representation that is compatible with the Lexical Markup Framework (LMF),[20] CMDI-compatible metadata[21] have been provided, and the data categories used in the database have been linked to data categories in the ISOCAT data category registry,[22] thus preparing its incorporation into the CLARIN research infrastructure and increasing the visibility, accessibility and the interoperability potential of the database. The graphical user interface has been adapted to work directly with this XML format. This version of DuELME can also be obtained via the Dutch HLT Agency.

12.6 Concluding Remarks

This paper has addressed problems that MWEs pose for NLP systems, more specifically the lack of large and rich formalised lexicons for multi-word expressions for use in NLP, and the lack of proper methods and tools to extend the lexicon of an NLP-system for multi-word expressions given a text corpus in a maximally automated manner. The paper has described innovative methods and tools for the automatic identification and lexical representation of multi-word expressions.

The identification methods operate on fully parsed sentences and can therefore use quite sophisticated manners of MWE identification that can abstract from different inflectional forms, differences in order, and deal with non-adjacent MWE components. Considerable progress has been achieved in this domain, and the

[18] http://www.tst-centrale.org/nl/producten/lexica/duelme/7-35

[19] http://www.clarin.nl

[20] http://www.lexicalmarkupframework.org/ and [4].

[21] CMDI stands for Components-based MetaData Infrastructure, cf. http://www.clarin.eu/cmdi and [1].

[22] http://www.isocat.org and [9].

methods developed have served as a basis for constructing a corpus-based lexical database for Dutch MWEs. However, there are still many opportunities for improvement. The most important topic for further research consists of finding methods for yielding more precise results for MWE identification, so that the manual step of selecting candidate MWEs can be significantly reduced.

The parameterised ECM has been investigated in detail on a large scale for Dutch. A full elaboration of the ECM parameters required for Dutch has been carried out. The incorporation method of the parameterised ECM has been tested in NLP systems, and an initial evaluation of the effect of MWEs incorporated in NLP systems has been carried out. Of course, there are opportunities for improvement here as well. It is in particular necessary to investigate in more detail how ECM parameters influence large scale integration of MWEs in NLP systems: to what extent can the parameters indeed be dealt with independently of the equivalence classes?

The paper describes a 5.000 entry corpus-based multi-word expression lexical database for Dutch developed using these methods. The database has been externally validated, and its usability has been evaluated in NLP-systems for Dutch. The MWE database developed fills a gap in existing lexical resources for Dutch. The generic methods and tools for MWE identification and lexical representation focus on Dutch, but they are largely language-independent and can also be used for other languages, new domains, and beyond this project. The research results and data described in this paper have therefore significantly contributed to strengthening the digital infrastructure for Dutch, and will continue to do so in the context of the CLARIN research infrastructure.

Acknowledgements The paper describes joint work by the IRME project team, and especially work carried out by Nicole Grégoire and Begoña Villada Moirón. I have liberally used material from reports, articles, PhDs etc. written by them, for which I am very grateful. I would also like to thank them and two anonymous reviewers for useful suggestions to improve this paper.

References

1. Broeder, D., Kemps-Snijders, M., Uytvanck, D.V., Windhouwer, M., Withers, P., Wittenburg, P., Zinn, C.: A data category registry- and component-based metadata framework. In: Calzolari, N., Maegaard, B., Mariani, J., Odijk, J., Choukri, K., Piperidis, S., Rosner, M., Tapias, D. (eds.) Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010), Valletta, pp. 43–47. European Language Resources Association (ELRA), Valletta (2010)
2. Broekhuis, H.: Het voorzetselvoorwerp. Nederlandse Taalkunde **9**(2), 97–131 (2004)
3. Copestake, A., Lambeau, F., Villavicencio, A., Bond, F., Baldwin, T., Sag, I., Flickinger, D.: Multiword expressions: linguistic precision and reusability. In: Proceedings of the 3rd

International Conference on Language Resources and Evaluation (LREC 2002), Las Palmas, pp. 1941–7. ELRA (2002)

4. Francopoulo, G., George, M., Calzolari, N., Monachini, M., Bel, N., Pet, M., Soria, C.: Lexical markup framework (LMF). In: Proceedings of LREC 2006, Genoa, pp. 233–236. ELRA, Genoa (2006)
5. Grégoire, N.: Untangling multiword expressions: a study on the representation and variation of Dutch multiword expressions. Phd, Utrecht University, Utrecht (2009). LOT Publication
6. Grégoire, N.: DuELME: A Dutch electronic lexicon of multiword expressions. J. Lang. Resour. Eval. **44**(1/2), 23–40 (2010). http://dx.doi.org/10.1007/s10579-009-9094-z
7. Hoekstra, H., Moortgat, M., Renmans, B., Schouppe, M., Schuurman, I., van der Wouden, T.: CGN syntactische annotatie. CGN report, Utrecht University, Utrecht (2003). http://lands.let.kun.nl/cgn/doc_Dutch/topics/version_1.0/annot/syntax/syn_prot.pdf
8. Hollebrandse, B.: Dutch light verb constructions. Master's thesis, Tilburg University, Tilburg (1993)
9. Kemps-Snijders, M., Windhouwer, M., Wright, S.: Principles of ISOcat, a data category registry (2010). Presentation at the RELISH Workshop Rendering Endangered Languages Lexicons Interoperable Through Standards Harmonization – Workshop on Lexicon Tools and Lexicon Standards, Nijmegen, 4–5 August 2010. http://www.mpi.nl/research/research-projects/language-archiving-technology/events/relish-workshop/program/ISOcat.pptx
10. Kilgarriff, A., Tugwell, D.: Word sketch: extraction & display of significant collocations for lexicography. In: Proceedings of the 39th ACL & 10th EACL workshop 'Collocation: Computational Extraction, Analysis and Exploitation', Toulouse, pp. 32–38. (2001)
11. Martin, W., Maks, I.: Referentie Bestand Nederlands: Documentatie. Report, TST Centrale (2005). http://www.tst-centrale.org/images/stories/producten/documentatie/rbn_documentatie_nl.pdf
12. Merlo, P., Leybold, M.: Automatic distinction of arguments and modifiers: the case of prepositional phrases. In: Proceedings of the Fifth Computational Natural Language Learning Workshop (CoNLL-2001), Toulouse, pp. 121–128 (2001)
13. Odijk, J.: A proposed standard for the lexical representation of idioms. In: Williams, G., Vessier, S. (eds.) EURALEX 2004 Proceedings, vol. I, pp. 153–164. Université de Bretagne Sud, Lorient (2004)
14. Odijk, J.: Reusable lexical representations for idioms. In: Lino, M.T., Xavier, M.F., Ferreira, F., Costa, R., Silva, R. (eds.) Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC-2004), III, Lisbon, pp. 903–906. ELRA, Lisbon (2004)
15. Rosetta, M.: Compositional Translation, Kluwer International Series in Engineering and Computer Science (Natural Language Processing and Machine Translation), vol. 273. Kluwer, Dordrecht (1994)
16. Sag, I.A., Baldwin, T., Bond, F., Copestake, A., Flickinger, D.: Multiword expressions: a pain in the neck for NLP. LinGO Working Paper 2001-03 (2001). http://lingo.stanford.edu/csli/pubs/WP-2001-03.ps.gz
17. Van De Cruys, T.: Semantic clustering in Dutch. In: Sima'an, K., de Rijke, M., Scha, R., van Son, R. (eds.) Proceedings of the Sixteenth Computational Linguistics in the Netherlands (CLIN), pp. 17–32. University of Amsterdam, Amsterdam (2006)
18. Van de Cruys, T., Villada Moirón, B.: Semantics-based multiword expression extraction. In: Grégoire, N., Evert, S., Kim, S. (eds.) Proceedings of the Workshop 'A Broader Perspective on Multiword Expressions', Prague, pp. 25–32. ACL, Prague (2007)
19. van der Beek, L., Bouma, G., van Noord, G.: Een brede computationele grammatica voor het Nederlands. Nederlandse Taalkunde **7**, 353–374 (2002)
20. van Noord, G.: At last parsing is now operational. In: Mertens, P., Fairon, C., Dister, A., Watrin, P. (eds.) TALN06 Verbum Ex Machina. Actes de la 13e conference sur le traitement automatique des langues naturelles, Leuven, pp. 20–42 (2006)
21. Villada Moirón, B.: Evaluation of a machine-learning algorithm for MWE identification. Decision trees. STEVIN-IRME Deliverable 1.3, Alfa-Informatica, Groningen (2006). http://www-uilots.let.uu.nl/irme/documentation/Deliverables/BVM_D1-3.pdf

22. Wermter, J., Hahn, U.: Collocation extraction based on modifiability statistics. In: Proceedings of COLING 2004, Geneva (2004)
23. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)