Lang Resources & Evaluation (2010) 44:23–39
DOI 10.1007/s10579-009-9094-z

DuELME: a Dutch electronic lexicon of multiword expressions

Nicole Grégoire

Published online: 1 August 2009

Abstract This article describes the design and implementation of a Dutch Electronic Lexicon of Multiword Expressions (DuELME). DuELME describes the core properties of over 5,000 Dutch multiword expressions. This article gives an overview of the decisions made in order to come to a standard lexical representation and discusses the description fields this representation comprises. We discuss the approach taken, which is innovative since it is based on the Equivalence Class Method (ECM). It is shown that introducing parameters to the ECM optimizes the method. The selection of the lexical entries and their properties is corpus-based. We describe the extraction of candidate expressions from corpora and discuss the selection criteria of the lexical entries. Moreover, we present the results of an evaluation of the standard representation in Alpino, a Dutch dependency parser.

Keywords Dutch · Lexicon · Multiword expressions

1 Introduction

This article describes the design and implementation of a Dutch Electronic Lexicon of Multiword Expressions (DuELME).[1] DuELME is one of the results of the project Identification and Representation of Multiword Expressions (IRME) and contains lexical descriptions of over 5,000 multiword expressions (MWEs). The lexical descriptions boast to be highly theory- and implementation-neutral. The lexicon is primarily intended for the use in various Dutch NLP systems.

[1] DuELME v1.0 has been validated by CST Copenhagen resulting in DuELME v1.1. The lexicon will be made available through the *TST-centrale* (HLT Agency, http://www.tst.inl.nl/).

N. Grégoire (✉)
UiL-OTS, University of Utrecht, Utrecht, The Netherlands
e-mail: n.h.w.gregoire@uu.nl

MWEs are known to be problematic for natural language processing. A considerable amount of research has been conducted in this area. Progress has been made especially in the field of multiword identification (Van de Cruys and Villada Moirón 2007; Fazly and Stevenson 2007). Moreover, interesting work has been done on the representation of MWEs by e.g. Dormeyer and Fischer (1998); Villavicencio et al. (2004), and Fellbaum et al. (2006).

Although our approach is in line with some of the projects cited, our work is also distinctive because (1) it is based on the Equivalence Class Method (ECM) (Odijk 2004); (2) the selection of the lexical entries and their properties is corpus-based; (3) it does not solely focus on one type of MWEs, but on MWEs in general; (4) the lexicon includes over 5,000 unique expressions; (5) it focuses on Dutch and is intended for use in NLP systems; and (6) a conversion to the Dutch NLP system Alpino[2] has been tested.

We took an innovative approach based on the ECM. The idea behind the ECM is that MWEs that have the same syntactic pattern require the same treatment in an NLP system. Accordingly, MWEs in DuELME are grouped on the basis of their pattern description. This method is really powerful since detailed pattern descriptions can be used for describing the characteristics of a group of MWEs.

Besides the description of the MWE patterns, we designed a uniform representation for the description of the individual expressions. Both the pattern descriptions and the MWE descriptions are implemented in DuELME.

The article starts with discussing the approach taken in Sect. 2. This is followed by describing the MWE extraction and selection procedure in Sect. 3. Section 4 elaborates on the representation of the MWEs and their patterns. An evaluation is described in Sect. 5. The article ends with a conclusion and discussion in Sect. 6.

2 Approach taken

2.1 Equivalence class method

An electronic resource that is meant to be used in various NLP systems should be organized in such a way that its integration into an NLP system can be done with a minimal amount of manual effort. The approach taken here is based on the Equivalence Class Method (Odijk 2004). Following the ECM, MWEs are grouped according to their syntactic pattern. MWEs with the same pattern form so-called Equivalence Classes (ECs). Having the ECs, representations for a specific theory and implementation can be derived. The procedure is that one instance of an EC must be converted in part manually. By defining and formalizing the conversion procedure, the other instances of the same EC can be converted fully automatically.

In the original approach, MWEs are grouped by syntactic pattern represented by a pattern identifier which is documented in a detailed pattern description. The pattern description not only includes the syntactic category of the head of the expression, the complements it takes and the description of the internal structure of

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

the complements, but also morpho-syntactic information of the individual components. An example of such a pattern description is given in (1).

(1) Expressions headed by a verb, taking a direct object consisting of a determiner and a singular noun.

Examples of MWEs that satisfy the description in (1) and together form an EC are *de stormbal hijsen* (lit. 'hoist the storm cone', id. 'to warn'), *de kar trekken* (lit. 'pull the car', id. 'carry the load') and *de boot missen* ('miss the boat').

A potential problem of the ECM as proposed is the risk that the number of ECs will run into thousands of which the majority contains only a small number of MWEs.[3] Since the ECM concentrates on minimizing the manual work when incorporating a large number of MWEs in a specific system, the method will be less successful if there are many ECs with only a few instances. In order to reduce the number of ECs and to increase the number of members within each EC, Odijk (2004) introduced the parameterized equivalence classes.

2.2 Parameterized equivalence class method

The central idea behind the parameterized ECM is that many MWE patterns describe structures that are for a large part identical and differ only locally. Pattern description (1) requires a singular noun, but another pattern is required that is identical except that it requires a plural noun. Moreover, another pattern is needed for a diminutive singular noun, and another one that requires a diminutive plural noun. In most theories and NLP systems such local differences are treated locally, e.g. locally different rule names (Rosetta 1994) or features. Odijk (2004) makes use of this fact by introducing parameters to represent local variation. Parameters are specified outside the pattern descriptions, i.e. in the parameterized ECM morpho-syntactic information of the individual components is not part of the pattern description. Instead of having a pattern description (1) for MWEs such as *de stormbal hijsen* and another pattern description (2) for MWEs such as *de benen nemen* (lit. 'to take the legs', id. 'to escape'), there is one pattern description (2) for both types of MWEs.

(2) Expressions headed by a verb, taking a direct object consisting of a determiner and a plural noun.
(3) Expressions headed by a verb, taking a direct object consisting of a determiner and noun.

Parameters are represented in the Component List (CL). The CL is part of the MWE description, see Sect. 4.3, and contains the obligatory lexically fixed components of an MWE in the canonical (or non-inflected) form. The term *parameter* is a feature and can be defined as an occurrence of the pair <parameter category, parameter value>, where *parameter category* refers to the aspect we parameterize, and *parameter value* to the value a parameter category takes. Examples of parameters are <nnum, sg> for singular nouns, <afrm, sup> for

[3] This problem was also raised by Copestake et al. (2002), though not in relation to the ECM.

superlative adjectives, <vfrm, part> for particle verbs. A total of 26 parameters have been defined for Dutch, see Grégoire (2007a) for an overview. Parameter values are notated between square brackets directly to the right of the item they parameterize, e.g. *de stormbal[sg] hijsen* and *de been[het][pl] nemen*.

Though extending the ECM with parameters introduces more theory-dependent assumptions, the approach as a whole is still as theory-neutral as possible: 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.

The extension with parameters contributes to reducing the number of ECs and increasing the number of members within each EC. As a result the number of MWEs that have to be dealt with manually decreases, whereas the number of MWEs that can be incorporated into an NLP system in a fully automatic manner increases. The successfulness of the method depends on (1) how many different ECs are distinguished (the less the better), and (2) how many instances each ECs contains (the more the better).

To determine the effectiveness of the method, measurements have been carried out on DuELME. A total of 5,232 unique expressions were included in the evaluation. To measure the number of ECs without parameters, we counted the number of unique parameter combinations from the CL-fields of each parameterized EC. For example, in the parameterized ECM the CLS *de stormbal[sg] hijsen* and *de been[het][pl] nemen* occur in the same EC. In the original ECM, these CLS would appear in different ECs, due to the variation of the number and of the gender of the noun.

Table 1 shows the major findings of the measurements. The first row, for example, means that 50% (or 2,616) of the expressions can be dealt with by 101 ECs in the original ECM and just 10 classes in the parameterized ECM. The main conclusion that can be drawn from the results is that introducing parameters in the ECM reduces the number of ECs by almost 90%, and multiplies the average cardinality of the ECs with a factor of over 9.4 for the whole set of MWEs.

To conclude, even though the successfulness of the method depends on the complexity of the incorporation of a parameter into a specific system, which varies from system to system, the additional effort is limited and counts for little compared to the reduction of manual effort that is gained by introducing the parameters.

Table 1 Coverage of ECs

Coverage (%)	# MWEs	# ECs	# Parameterized 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	139

3 Data

The selection of the lexical entries and their properties is corpus-based. The use of corpora is necessary but not sufficient. It is necessary because we want our lexicon to reflect actual language usage and because we do not want to restrict ourselves to a linguist's imagination of which uses are possible or actually occur. On the other hand, using corpora to extract the MWEs is not sufficient for the following reasons: (1) the techniques sometimes erroneously identify groups of words as an MWE; (2) the extraction techniques sometimes group different expressions that share some but not all words together; and (3) the extraction is in part based on an automatic syntactic parse of the corpus sentences, and these parses may be incorrect.[4] Because of the unreliable output, the data extracted were carefully analyzed before creating entries for MWEs.

Section 3.1 addresses the extraction of the data from corpora, and Sect. 3.2 elaborates on the selection of MWEs for DuELME.

3.1 Extraction[5]

The candidate expressions[6] for DuELME are extracted from the Dutch CLEF corpus, a collection of newspaper articles from 1994 to 1995, taken from the Dutch daily newspapers Algemeen Dagblad and NRC Handelsblad. The corpus contains 80 million words and 4 million sentences, which have been annotated automatically with the Alpino parser.

The automated extraction of MWEs requires predefined patterns. We created a list of patterns on the basis of a random selection of MWEs taken from the Van Dale Lexical Information System (VLIS) database and chose the five most frequently occurring patterns, shown in (1). The patterns have been used as defined, i.e. the patterns do not include any other complements than the ones stated.

(4)
NP_V NP(DIRECT OBJECT)–verb
(NP)_PP_V variable NP(DIRECT OBJECT)–PP–verb
NP_NP_V NP(INDIRECT OBJ.)–NP(DIRECT OBJECT)–verb
A_N adjective–noun
N_PP noun–PP
P_N_P preposition–noun–preposition

The tuples, i.e. sequences of lemmas formed by the head of the pattern and the heads of the complements, extracted from the corpus form the input for the

[4] Furthermore, automatic extraction techniques fail to come up with all the MWEs that occur in the corpora (Villada Moirón 2007a). However, this problem cannot be overcome by manually checking the automatically extracted data.

[5] The identification of MWEs has been done by Begoña Villada Moirón working at the University of Groningen.

[6] For convenience we speak of *candidate expressions*, in practice, the expressions extracted from the corpus are actual lemma pairs, triples or quadruples, i.e. combinations of two, three or four words, depending on the pattern of the extracted data, that may form an MWE or may be part of an MWE.

Table 2 Distribution of candidate expressions over the extracted patterns

Pattern	# of candidate expressions
NP_V	3,894
(NP)_PP_V	2,405
NP_ NP_V	202
A_N	1,001
N_PP	1,342
P_N_P	607
Total	9,451

identification models. Based on experiments with various machine learning techniques, Villada Moirón (2006) chose to apply a decision tree classifier. The decision tree classifier proposes a class (MWE|noMWE) for each input tuple. The identification provides a list of candidate expressions, i.e. tuples that are assigned the class *MWE*, yielding a total of 9,451 expressions, see Table 2. No manual filtering or correction has been applied to this list at this stage.

MWEs allow morpho-syntactic variation, e.g. verbs may show different forms depending on tense, person, etc.; nouns may allow number alternation, etc. Evidence of morpho-syntactic variation has been collected from the Twente Nieuws Corpus (TwNC) (Ordelman 2002). The TwNC comprises 500 million words of newspaper text and television news reports. The corpus has also been syntactically annotated with the Alpino parser. For each candidate expression a set of properties has been extracted, see Sect. 3.2 for an example.

3.2 Selection

The candidate expressions, their properties and example sentences form the input for the data selection. The MWEs for the lexicon have been selected according to the definition given in (2).

(5) A multiword expression is a combination of words that has linguistic properties not predictable from the individual components or the normal way they are combined.

Examples of such linguistic properties are:

– Lexical properties: specific lexical items must be used and cannot be replaced by synonyms or near-synonyms without changing the meaning or the well-formedness of the expression. Two Dutch examples are:

(6) blunder maken/begaan/*doen/*slaan
mistake make/commit/*do/*hit
'make a mistake'

(7) flater *maken/*begaan/*doen/slaan
mistake *make/*commit/*do/hit
'make a mistake'

- Morphological properties, e.g. *e*-inflection on the noun: *ten gevolg**e** van* (‘because of’).
- Syntactic properties, e.g. the lack of a determiner preceding a singular count noun, which is in general prohibited in standard Dutch grammar: *in opdracht van* (‘by order of’).
- Semantic properties: the meaning of the expression cannot be deduced from the meaning of the individual components, e.g.:

(8) met de handen in het haar zitten
with the hands in the hair sit
‘to be at loss what to do’

The morphological, syntactic and semantic properties of an analysed expression often lead to a clear decision of whether the expression is a true MWE. Deciding whether a combination is a true MWE solely on the basis of its lexical properties is not always as clear-cut, especially not for direct object–verb combinations, since in many cases not all properties of the individual components or the normal rules to combine them are known.

An example of a clear MWE is *een gesprek voeren* (‘have a conversation’): although one meaning of *voeren* is “being actively occupied with”, and although one can be actively occupied with a conversation, the combination is unpredictable since *gesprek* cannot be substituted by its synonym *praatje* (‘chat’), i.e. *een praatje voeren* (‘have a chat’) is out. For this reason, *een gesprek voeren* is a true MWE and thus entered in the lexicon.

An illustration of a not so clear-cut example is the expression *een getuigenis afleggen* (‘give a testimony’). The extracted data contain five other nouns that occur with *afleggen*, three of which requiring the same meaning of *afleggen* as required by the noun *getuigenis*: *verklaring* (‘statement/testimony’), *eed* (‘oath’), and *bekentenis* (‘confession’). The question is whether the lexical selection of the noun is predictable according to its semantic properties. In this case we are not sure, since we do not know which semantic properties a noun that selects the verb *afleggen* requires. Although the expression seems semantically regular, resource constraints prevented us from conducting a detailed study for each of such cases and forced us to make a pragmatic decision on this point. Concretely this means that in this case all four expressions have been included in the lexicon.

A single data record from the extracted data may contain a lemma tuple that is part of more than one MWE. An example of such a data record is given in Table 3.[7]

The tuple is *hand hebben*, and given the extracted properties, the example sentences and language knowledge, at least four different expressions can be identified:[8]

[7] The numbers represent the absolute frequency of the number of occurrences of the value.

[8] As stated, the extracted pattern does not include any other complements than the ones defined. In this case the extracted pattern is direct object–verb. Given the example sentences we can conclude that the Alpino parser analyzes PPs as modifiers instead of complements, because the subcategorization pattern of *hebben* (‘have’) individually differs from the subcategorization pattern of *hebben* in the expressions *the hand hebben in iets* and *de handen vol hebben aan iets*, cf. 234.xml and 452.xml.

Table 3 Example of a data record

Property	Value
tuple	heb#hand
frame	transitive_ndev 1280,np_ld_pp 181, aci_simple 22,
frequency	1,497
head	heb
subject	hij 149,die 96,ik 70,ze 67,je 46,zij 28,we 27,
complement	hand
head of complement	hand
dependency	obj1 1497,
number	sg 908,pl 589,
diminutive	nodim 1497,
determiner	de 696,een 235,NO 208,geen 90,zijn 74,hun 62,
premodifier	NO 875,gelukkig 123,vrij 118,schoon 75,
postmodifier	NO 1186,in 115,van 99,op 24,bij 14,vol 12,
examples	hij had zijn handen vol om een boterham te verdienen
	en heeft de handen vol aan drugssmokkelaars
	Hij is een pianist die vier handen leek te hebben
	Het Iraakse regime heeft de hand gehad in de dood van
	Ook daar had God de hand in
	De meisjes hadden hun handen op de gebogen knieën

(9) a. de vrije hand hebben
the free hand have
'have a free hand'
b. een gelukkige hand hebben
a lucky hand have
'be lucky'
c. de hand hebben in iets
the hand have in sth.
'have a hand in sth.'
d. de handen vol hebben aan iets
the hands full have on sth.
'have one's hands full with sth.'

Solely the head of the predefined pattern and the heads of the complements have been taken into account with the automated extraction, i.e. no explicit search has been done for e.g. adjectives modifying the head of the direct object. Combinations such as determiner-adjective-noun have been created and checked manually using the extracted properties, the example sentences, language knowledge and in some

cases a dictionary. For this reasons DuELME contains a total of 141 MWE patterns, while solely five patterns have been used as input for the automated extraction.

To summarize, MWEs for the lexicon are selected from lists of candidate expressions, their properties and example sentences according to the definition given in (2). The selection needs to be done manually, since there is no straightforward way to interpret the data fully automatically. The information given in the data record needs to be analyzed carefully to identify one or more MWEs and to determine the correct form of an MWE.

4 Representation

Various aspects played a role in the representation used in DuELME. The main requirement of the standard encoding is that it can be converted into any system specific representation with a minimal amount of manual work. The method adopted to achieve this goal is the Equivalence Class Method, discussed Sect. 2. In order to form equivalence classes, DuELME contains besides MWE descriptions also MWE pattern descriptions. For the development of the representation two Dutch parsers have been consulted, viz. the Alpino parser and the Rosetta MT system (Rosetta 1994). The description of an MWE consists of a list of core properties specific for a certain MWE and a pattern name that refers to the description of an MWE pattern.

Ideally, an MWE description contains besides basic lexical information also semantic information and detailed syntactic information, such as to which extent an MWE can undergo certain syntactic transformations. Except for modifiability, no syntactic operations are included in the description of MWEs in DuELME. Besides the fact that proof for the presence of syntactic variability of an MWE is often hard to find, we decided to describe only a number of core properties of MWEs because of resource limitations. We are confident that this resource can form a good basis for an even more complete description of MWEs.

In an earlier version of DuELME, each MWE was classified as either fixed, semi-fixed or flexible. Section 4.1 addresses the reasons why we discontinued this classification. The MWE pattern description is discussed in Sect. 4.2 and the MWE description is elaborated in Sect. 4.3. Detailed information about the ingredients that are part of the descriptions can be found in (Grégoire 2007a).

4.1 Subclasses revised

As stated, in an earlier version of DuELME, MWEs were classified as either fixed, semi-fixed or flexible. In general, this classification conforms to the classification given in a well-known paper on subclasses written by Sag et al. (2001). Sag et al. make a distinction between *lexicalized phrases* and *institutionalized phrases*. Lexicalized phrases are subdivided into fixed, semi-fixed and flexible expressions. The most important reason for this subdivision is the variation in the degree of syntactic flexibility of MWEs. Roughly they claim that syntactic flexibility is related to semantic decomposability. Semantically non-decomposable idioms are idioms the meaning of which cannot be distributed over its parts and which are therefore

not subject to syntactic variability. Sag et al. state that "the only types of lexical variation observable in non-decomposable idioms are inflection (*kicked the bucket*) and variation in reflexive form (*wet oneself*)." Examples of non-decomposable idioms are the oft-cited *kick the bucket* and *shoot the breeze*. On the contrary, semantically decomposable idioms, such as *spill the beans*, tend to be syntactically flexible to some degree.

Although this classification might work for simple constructions such as direct object–verb combinations, it becomes more difficult to categorize MWEs where the verb takes two arguments. Take for example the modifiability of expressions. One of the characteristics of semi-fixed expressions is that the expression can be modified as a whole, while the main characteristic of flexible MWEs is the fact that also the individual components within the MWE can be modified. The classification may work to account for the differences in modifiability between the expressions *de stormbal hijsen*, which would be classified as semi-fixed, and *een bok schieten* (lit. 'to shoot a male goat', id. 'to make a blunder'), which would be classified as flexible, but given the expression *olie op het vuur gooien* ('add fuel to the fire'), *olie* can be modified, e.g. by *extra* or *nieuw* ('new'), but *vuur* cannot be modified (without loosing the idiomatic meaning of the expression). It is not possible to characterize this expression as either semi-fixed or flexible.

Revising the use of subclasses, we came to the conclusion that applying such a classification would complicate the representation and not enrich it. Besides the fact that problems arise with MWEs that include a verb that takes more than one argument, a disadvantage of the classification is that the subclasses solely distinguish between modifiable and unmodifiable, while the data show that a noun can also be limited modifiable, i.e. it is not freely modifiable nor unmodifiable. Instead of using classes to describe an MWE, we start from the basic principle that every MWE can be modified as a whole, and we describe the modifiability of each individual component in the MWE pattern description.

4.2 MWE pattern description

As stated, expressions are classified according to their pattern. In the original ECM the pattern is an identifier which refers to the structure of the MWE represented as free text in which the uniqueness of the pattern is described. This description includes the syntactic category of the head of the expression, the complements it takes and the description of the internal structure of the complements. Furthermore it is described whether individual components can be modified. In the current approach, a formal representation of the patterns has been added to the pattern descriptions, see (10).

(10) Expressions headed by a verb, taking a direct object consisting of a fixed determiner and an unmodifiable noun.
[.VP [.obj1:NP [.det:D (1)] [.hd:N (2)]] [.hd:V (3)]]

Since this formal representation is in agreement with a de facto standard for Dutch (van Noord et al. 2006), most Dutch NLP systems are able to use it for the conversion procedure, yielding an optimal reduction of manual labor.

Table 4 Additional labels to cover modifiability of nouns and adjectives

Label	Description
A	Not modifiable adjective
A1	Modifiable adjective
N	Not freely modifiable noun
N1	Modifiable noun
N2	Limited modifiable noun

The notation used to describe the patterns is a formalization of dependency trees, in particular CGN (*Corpus Gesproken Nederlands* 'Corpus of Spoken Dutch') dependency trees (Hoekstra et al. 2003). CGN dependency structures are based on traditional syntactic analysis described in the Algemene Nederlandse Spraakkunst (Haeseryn et al. 1997) and are aimed to be as theory neutral as possible.

The patterns are encoded using a formal language, which is short and which allows easy visualization of dependency trees. The dependency labels (in lower case) and category labels (in upper case) are divided by a colon (:), e.g. *obj1:NP*. For leaf nodes, the part-of-speech is represented instead of the category label. To cover the modifiability of the noun and adjective,[9] additional labels have been created, see Table 4.

It should be noted that often it is not clear whether a noun is limited modifiable or freely modifiable, and whether the limited modifiability of the noun is the result of the combination of the noun with the other components of the expression or that it is a property of the noun itself. The determination of whether the noun is limited modifiable or freely modifiable is merely based on corpus information, which may not be exhaustive and may lead to an incorrect pattern allocation.

Leaf nodes are followed by an index that refers to the MWE component as represented in the CL-field (see Sect. 4.3), e.g. (1) refers to the first component of the CL, (2) to the second, etc. Variables are represented similar to the indices of MWE components, e.g. [obj1:NP (var)], [obj2:NP (var)], etc.:

(11) *iemand de helpende hand bieden*
(lit. 'offer s.o. the helping hand', id. 'lend s.o. a hand')
[.VP [.obj2:NP (var)] [.obj1:NP [.det:D (1)] [.mod:A (2)] [.hd:N (3)]] [.hd:V (4)]]

The pattern is part of the MWE pattern description which includes, besides a pattern name, a pattern and a textual description, four additional fields, viz.:

pos encodes the part-of-speech tag for each leaf node in the PATTERN-field. The POS-field is mainly used for maintenance reasons, i.e. with the help of this field it is possible to limit the number of candidate pattern descriptions for an expression.
mapping indicates the relation between the position of a component in the Component List (CL) and its position in the EXAMPLE-field, i.e. the relation between non-inflected forms and full forms, see Sect. 4.3.

[9] Modifiability of the adjective includes variation of the form, e.g. comparative and superlative.

Table 5 Example of an MWE pattern description

PATTERN_NAME	ec7
POS	d n v
PATTERN	[.VP [.obj1:NP [.det:D (1)] [.hd:N1 (2)]] [.hd:V (3)]]
MAPPING	3 4 5
EXAMPLE_MWE	zijn debuut maken
EXAMPLE_SENTENCE	hij heeft zijn debuut gemaakt
DESCRIPTION	Expressions headed by a verb, taking a direct object consisting of a fixed determiner and a modifiable noun.
COMMENT	

example_mwe contains an example of how to represent the MWE in the EXPRESSION-field of the MWE description.
example_sentence illustrates how to represent the example sentence of an MWE in the EXAMPLE-field of the MWE description.
comment which can be used to specify notes.

An example of an MWE pattern description stored in DuELME is given in Table 5.

4.3 MWE description

In addition to the MWE pattern descriptions, the lexicon contains MWE descriptions. The description of an MWE consists of two parts, viz. a basic MWE description and an additional MWE description.

The basic MWE description comprises six fields, see Table 6 for two examples.

expression contains the obligatory lexically fixed components of an MWE in the full form.
cl The Component List contains the same components as the EXPRESSION-field. The difference is that the components in the CL are in the canonical (or non-inflected) form, instead of in the full form. Parameters are used to specify the full form characteristics of each component, see Sect. 2.2.
pattern_name is used to assign an MWE pattern description to the expression. Up to three patterns can be specified for each MWE. An example of an entry with multiple patterns represented is *college geven* ('lecture'): the assignment of

Table 6 Two examples of basic MWE descriptions

EXPRESSION	zijn kansen waarnemen ('to seize the oppurtunity')	blunder ('mistake')
CL	zijn kans[pl] waar_nemen[part]	blunder
PATTERN_NAME	ec1	ec2
LISTA	n.a.	maken ('make')
LISTB	n.a.	begaan ('commit')
EXAMPLE	hij heeft zijn kansen waargenomen	hij heeft een blunder begaan

PATTERN_NAME1 yields the MWE *college geven*, and the assignment of PATTERN_NAME2 yields the MWE *college geven aan iemand* ('lecture s.o.').

lista and listb The use of these fields is restricted to three types of expressions:

- Combinations of a verb that seems to have very little semantic content and a prepositional phrase, a noun phrase or an adjectival phrase. Since the complement of the verb is used in its normal sense, the constructions are subject to standard grammar rules, which include passivization, internal modification, etc.
- Combinations of a noun and a verb that may be a regular combination, but since the exact properties of the individual components are unknown, the combination is treated as an MWE.
- Combinations of an adjective with an irregular meaning and a noun that is used in its literal sense, e.g. *zwaar accent* ('strong accent').

The lexical selection of the verb and the adjective is highly restricted, but not always limited to one. The alternation of the verb or the adjective should be specified in the LIST-fields. The reason for using two LIST-fields is to separate predefined list values from special list values. The predefined list values are high-frequency verbs that are known to occur often as so-called light verbs, especially with PPs. Two sets of verbs are predefined:

1. *blijken* ('appear') *blijven* ('remain') *gaan* ('go') *komen* ('come') *lijken* ('appear') *raken* ('get') *vallen* ('fall')[10] *worden* ('become') *zijn* ('be')
2. *brengen* ('bring') *doen* ('do') *geven* ('give') *hebben* ('have') *houden* ('keep') *krijgen* ('get') *maken* ('make') *zetten* ('put')

A complement co-occurs either with verbs from set A or with verbs from set B. Each verb from the chosen set is checked against the occurrences found in the corpus data. If a verb does not occur in the corpus data and also not in self-constructed data, it is deleted from the LISTA-field. The LISTB-field contains lexemes, either verbs or adjectives, that are not in the predefined set but do co-occur with the component(s) in the EXPRESSION-field. The information in the LISTB-field is merely based on corpus data and therefore may not be exhaustive.

example contains an example sentence with the expression. The only requirement of this field is that its structure is identical for each expression with the same PATTERN_NAME.

The additional MWE description contains the following fields:

subject is used to cover subject restrictions and can contain both a list of heads of possible subjects extracted from annotated corpora and predefined labels such as [sg] for singular subject.

object is used to cover object restrictions and can contain both a list of heads of possible objects extracted from annotated corpora and predefined labels such as [anim] for animate object.

[10] The literal meaning of *vallen* is 'fall', but it has a variety of different meanings in MWEs of this type, including 'become', 'is experienced as', etc.

modifier is used to list modifiers (including adjectives modifying a noun). In the current encoding this field is mainly filled with modifiers coming from extracted data.
rpron is used to encode pronominalized PP realizations, and contains either the predefined label [ssub] for realizing the complement of the pronominalized PP as a clause starting with a complementizer, or the label [vp] for realizing the complement of the pronominalized PP as an infinitive clause.
conjugation is used to specify whether the head of the expression conjugates with *zijn* ('to be'), or *hebben* ('to have'), or both.
polarity is *none* by default and takes the value *NPI* (Negative Polarity Item) if an expression can only occur in negative environments, and *PPI* (Positive Polarity Item) if an expression can only occur in positive environments.

Furthermore, the MWE description contains a field with a reference to a plain text file in which the information extracted from the corpora is stored.

It must be noted that the main focus is on representing those properties that are needed for a successful implementation of the MWE lexicon in any specific NLP system. This means that the priority is on properly describing the fields that are part of the basic MWE description, and although the additional description fields also form an important part of the description, it cannot be guaranteed that these fields are completely filled or free from errors. Any comments regarding the MWE description are entered in the optional COMMENT-field.

5 Evaluation

DuELME has been evaluated by testing whether it can be successfully used for the purpose it was developed for, viz. the semi-automatic incorporation of the lexical representations into NLP systems. We extensively studied the way the Rosetta MT system (Rosetta 1994) deals with MWEs and moreover what is needed for the incorporation of the standard in Rosetta. A conversion procedure has been described in detail in Grégoire (2007c), but could unfortunately not be tested in practice. The incorporation of a part of DuELME into Alpino has been tested in theory and in practice.

Alpino is a dependency parser for Dutch, which uses linguistic knowledge and various heuristics to construct appropriate linguistic structures of Dutch sentences. The incorporation of DuELME in Alpino comprises adding new lexical entries to the Alpino lexicon. For the purpose of the test, we left the Alpino grammar untouched. Therefore only types of MWE constructions that are already present in the Alpino lexicon can be integrated.

We have converted the standard representation following the spirit of the ECM, viz. take one instance from an EC, define and formalize the conversion of this instance, and use the information gathered to automate the conversion of all other instances of the same EC. The output of the semi-automatic conversion is basically a new lexicon that includes the original Alpino lexicon extended with the verbal

Table 7 CA scores

Lexicon	CA
Alpino lexicon	82.8
Extended lexicon	94.1

MWEs from DuELME. The implementation of DuELME in Alpino has been described exhaustively in Grégoire (2007b).

The assessment of the effect of incorporating the standard into Alpino has been reported in Villada Moirón (2007b). The evaluation that has been carried out is rather small but nonetheless promising. A sample of 100 sentences with an MWE extracted from DuELME has been used to test the accuracy of the parser for both the original Alpino lexicon and the Alpino lexicon extended with verbal MWEs from DuELME. The sentences have been assigned a manually created parse to serve as a reference parse for the evaluation.

The sentences have been parsed both with the original Alpino lexicon and with the extended lexicon. Given that the extended lexicon contains more lexical entries for MWEs, it is expected that when Alpino uses the extended lexicon, more sentences with MWEs are correctly analysed than when Alpino uses the original lexicon.

To measure the accuracy of the analyses returned by the parser, the *concept accuracy per sentence* (CA) has been computed as proposed in van Noord (2006) by comparing the parsed sentences with the manually created reference parses. The higher the concept accuracy the better the performance of the parser. Table 7 shows the concept accuracy per sentence for the set of MWE sample sentences using two different lexica. As expected, the results show that the concept accuracy of sentences that contain an MWE improves substantially when using the extended lexicon. For a more detailed description of the method and an overview of quantitative results see Villada Moirón (2007b).

6 Conclusion and discussion

We have given an overview of the decisions made in order to come to a standard lexical representation for Dutch MWEs and discussed the description fields this representation comprises. The strength of our method lies in the ability of grouping individual expressions according to their pattern, yielding multiple classes of MWEs. The advantage of creating classes of MWEs is that it eases the conversion of the standard representation into any system specific representation.

It was shown that introducing parameters to the ECM decreases the number of equivalence classes needed with almost 90% with respect to the numbers of equivalence classes needed in the original ECM. The ability to handle parameters varies from system to system, which means that some systems will profit more from the parameterized ECM than other systems.

MWEs for the lexicon have been selected from corpus-based lists of candidate expressions, their properties and example sentences. The integration of acquired lexical data in DuELME needs to be done manually, since there is no

straightforward way to interpret the data automatically. The information given in a data record needs to be analyzed carefully to identify one or more MWEs and to determine the correct form of an MWE.

We have created a resource that is suited for a wide variety of MWEs. The resource describes a set of essential properties of over 5,000 unique expressions. The set of properties can surely be extended, but we have limited ourselves to a number of core properties because of resource limitations. We are confident that this resource can form a good basis for an even more complete description of MWEs.

Acknowledgements The IRME project has been carried out within the STEVIN programme which is funded by the Dutch and Flemish Governments. The work on DuELME v1.1 has been completed in September 2007. The author would like to thank Jan Odijk for his valuable input to this article.

References

Copestake, A., Lambeau, F., Villavicencio, A., Bond, F., Baldwin, T., Sag, I., & Flickinger, D. (2002). Multiword expressions: Linguistic precision and reusability. In M. G. Rodríguez & C. P. S. Araujo (Eds.), *Proceedings of the 3rd international conference on language resources and evaluation (LREC 2002)* (pp. 1941–1947). Las Palmas, Spain.

Dormeyer, R., & Fischer, I. (1998). Building lexicons out of a database for idioms. In A. Rubio, N. Gallardo, R. Castro, & A. Tejada (Eds.), *Proceedings of the 1st international conference on language resources and evaluation* (pp. 833–838). Granada, Spain.

Fazly, A., & Stevenson, S. (2007). Distinguishing subtypes of multiword expressions using linguistically-motivated statistical measures. In N. Grégoire, S. Evert & B. Krenn (Eds.), *Proceedings of the ACL 2007 workshop on a broader perspective on multiword expressions* (pp. 9–16). Prague, Czech Republic.

Fellbaum, C., Geyken, A., Herold, A., Koerner, F., & Neumann, G. (2006). Corpus-based studies of German idioms and light verbs. *International Journal of Lexicography, 19*(4), 349–361.

Grégoire, N. (2007a). Design and implementation of a lexicon of Dutch multiword expressions. In N. Grégoire, S. Evert & B. Krenn (Eds.), *Proceedings of the ACL 2007 workshop on a broader perspective on multiword expressions* (pp. 17–24). Prague, Czech Republic.

Grégoire, N. (2007b). MWE lexicon for Dutch: Alpino conversion. Internal report published on http://www.uilots.let.uu.nl/irme/, STEVIN IRME, Utrecht, The Netherlands.

Grégoire, N. (2007c). MWE lexicon for Dutch: Rosetta conversion. Internal report published on http://www.uilots.let.uu.nl/irme/, STEVIN IRME, Utrecht, The Netherlands.

Haeseryn, W., Romijn, K., Geerts, G., de Rooij, J., & van den Toorn, M. (1997). *Algemene Nederlandse Spraakkunst*. Groningen en Deurne: Martinus Nijhoff and Wolters Plantyn.

Hoekstra, H., Moortgat, M., Renmans, B., Schouppe, M., Schuurman, I., & van der Wouden, T. (2003). CGN Syntactische Annotatie. Published on: http://lands.let.kun.nl/cgn/doc_Dutch/topics/version_1.0/annot/syntax/syn_prot.pdf, Utrecht, The Netherlands.

Odijk, J. (2004). A proposed standard for the lexical representation of idioms. In G. Williams & S. Vessier (Eds.), *EURALEX 2004 proceedings* (pp. 153–164). Lorient, France.

Ordelman, R. (2002). Twente Nieuws Corpus (TwNC). Parlevink Language Technology Group, Twente University.

Rosetta, M. T. (1994). *Compositional translation*. Dordrecht: Kluwer Academic Publishers.

Sag, I. A., Baldwin, T., Bond, F., Copestake, A., & Flickinger, D. (2001). Multiword expressions: A pain in the neck for NLP. In A. F. Gelbukh (Ed.), *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2002)* (pp. 1–15). Mexico City, Mexico.

Van de Cruys, T., & Villada Moirón, B. N. (2007). Semantics-based multiword expression extraction. In N. Grégoire, S. Evert & B. Krenn (Eds.), *Proceedings of the ACL 2007 workshop on a broader perspective on multiword expressions* (pp. 25–32). Prague, Czech Republic.

van Noord, G. (2006). At last parsing is now operational. In P. Mertens, C. Fairon, A. Dister, & P. Watrin (Eds.), *TALN06 Verbum Ex Machina. Actes de la 13e conference sur le traitement auto-matique des langues naturelles* (pp. 20–42). Leuven, Belgium.

van Noord, G., Schuurman, I., & Vandeghinste V. (2006). Syntactic annotation of large corpora in STEVIN. In N. Calzolari & K. Choukri (Eds.), *Proceedings of the fifth international conference on language resources and evaluation (LREC 2006)* (pp. 1811–1814). Genoa, Italy.

Villada Moirón, B. (2006). Evaluation of a machine learning algorithm for MWE identification. Decision trees. Internal report published on http://www.uilots.let.uu.nl/irme/, STEVIN IRME, Utrecht, The Netherlands.

Villada Moirón, B. (2007a). Identification and representation of multiword expressions. Poster presented at STEVIN-dag 2007. Hoeven: The Netherlands.

Villada Moirón, B. (2007b). A task-based evaluation of the ECM database. Effect on parsing performance. Internal report published on http://www.uilots.let.uu.nl/irme/, STEVIN IRME, Utrecht, The Netherlands.

Villavicencio, A., Copestake, A., Waldron, B., & Lambeau, F. (2004). The lexical encoding of MWEs. In T. Tanaka, A. Villavicencio, F. Bond, & A. Korhonen (Eds.), *Proceedings of the ACL 2004 workshop on multiword expressions: Integrating processing* (pp. 80–87). Barcelona, Spain.