This is a pre-print of an article published in Computational and Corpus-Based Phraseology.
The final authenticated version is available online at: https://doi.org/10.1007/978-3-319-69805-2_6

Putting the Horses before the Cart: Identifying Multiword Expressions before Translation

Carlos Ramisch[⋆]

Aix Marseille Univ, CNRS, LIF, Marseille, France
carlos.ramisch@lif.univ-mrs.fr

Abstract. Translating multiword expressions (MWEs) is notoriously difficult. Part of the challenge stems from the analysis of non-compositional expressions in source texts, preventing literal translation. Therefore, before translating them, it is crucial to locate MWEs in the source text. We would be putting the cart before the horses if we tried to translate MWEs before ensuring that they are correctly identified in the source text. This paper discusses the current state of affairs in automatic MWE identification, covering rule-based methods and sequence taggers. While MWE identification is not a solved problem, significant advances have been made in the recent years. Hence, we can hope that MWE identification can be integrated into MT in the near future, thus avoiding clumsy translations that have often been mocked and used to motivate the urgent need for better MWE processing.

1 Introduction

Translation is probably one of the most complex tasks in language processing, both for humans and computers. One of the reasons why translation is challenging is the arbitrary and non-categorical nature of human languages. In other words, while general grammatical and semantic composition rules are useful abstractions to model languages in computer systems, actual language use is permeated by exceptions that are often at the root of errors in language technology. Multiword expressions (MWEs) represent such exceptions to general language rules when words come together. They can be defined as combinations of at least two lexemes which present some idiosyncrasy, that is, some deviation with respect to usual composition rules at some level of linguistic processing [2]. Therefore, their automatic processing is seen as a challenge for natural language processing (NLP) systems [30, 5, 33].

If MWEs are a pain in the neck for language technology in general [30], this is especially true for machine translation (MT) systems. The automatic translation of MWEs by current MT systems is often used as a compelling argument for the importance of dealing with them in NLP systems [21, 24, 38]. For example, the two sentences below in English (EN) and in French (FR) contain an equivalent multiword expression which means carrying out a task with precipitation, in the wrong order, by inverting priorities:

- EN: *He puts the cart before the horses.*
- FR: *Il met la charrue avant les bœufs.*

[⋆] I would like to thank the chairs of MUMTTT 2017 for inviting me to the event and for giving me the oportunity to publish this invited contribution. This paper includes materials published in other venues and co-written with: Mathieu Constant, Silvio Cordeiro, Benoit Favre, Marco Idiart, Gülşen Eryiğit, Johanna Monti, Lonneke van der Plas, Michael Rosner, Manon Scholivet, Amalia Todirascu, and Aline Villavicencio. Work reported here has been partly funded by projects PARSEME (Cost Action IC1207), PARSEME-FR (ANR-14-CERA-0001), and AIM-WEST (FAPERGS-INRIA 1706-2551/13-7).

This is a pre-print of an article published in Computational and Corpus-Based Phraseology.
The final authenticated version is available online at: https://doi.org/10.1007/978-3-319-69805-2_6

While the FR expression is equivalent in meaning to the EN one, it translates word-for-word into EN as *He puts the plough before the oxen.* As a consequence, even though the automatic translation succeeds in translating the individual words, the translation of the whole expression fails, as we show in the examples below:[1]

- EN$\overset{MT}{\to}$FR: *Il met le chariot devant les chevaux.*
- FR$\overset{MT}{\to}$EN: *He puts the cart before the oxen.*

MT can be seen as a process of analysis and generation, that is, a source text is first *analysed* to create an abstract intermediate representation of its meaning, and then a target text is *generated* from this abstract representation so that the meaning of the source text is preserved in the target text [43]. Even though modern MT systems do not always explicitly model translation using Vauquois' triangle, the analysis/generation model is useful to understand the role of MWEs in MT. That is, MWE processing for MT means not only analysing them and getting their meaning correctly, but also generating them in the target text to ensure fluency and naturalness.

We focus only on the first step of translation, that is source text *analysis*, and on the role of MWE identification in the analysis step of MT. While generation is also important to confer naturalness to the output of the system, most research contributions to date in the MWE community have focused on text analysis, and work investigating MWE-aware text generation is quite rare. Therefore, we will explore the landscape of existing monolingual MWE identification methods that could be useful for MT.

This paper gathers methods and experimental results on MWE identification previously published in collaboration with colleagues (see the acknowledgements). Its structure is based on a survey on MWE processing [8], which distinguishes rule-based and statistical tagging methods. First, we briefly list and exemplify resources required and useful for MWE identification (Sec. 2). Then, we summarise previously published models for rule-based MWE identification (Sec. 3) and for sequence-tagging MWE identification (Sec. 4). We conclude by discussing the applicability of these systems as preprocessing step for MT, and perspectives for future work in the field (Sec. 5).

2 MWE Identification Resources

Automatic MWE identification is a task that consists in finding MWEs in running text, on the level of word occurrences or tokens. Figure 1, taken from [26], shows an example of sentence, with MWEs annotated in bold and additionally containing a category label on the last token. Notice that we use the term *identification* referring to in-context MWE identification, as opposed to MWE *discovery*, where the goal is to extract MWEs from text and include them in lexicons, as exained in [8]. Both tasks are similar, being given as input text where MWEs should be located. However, they differ in their output: while discovery generates MWE lists, identification generates annotations on the input sentences. Often MWE discovery can be considered as a prerequisite for identification, as the latter usually relies on lexicons built with the help of corpus-based MWE discovery.

Identification methods take text as input and, in order to locate MWEs, also require additional information to guide the process. This additional information is of two types: (a) more or less sophisticated *lexicons* containing MWE entries and sometimes contextual information about their occurrences, and (b) *probabilistic models* learned using machine learning methods applied to corpora where MWEs were manually annotated. In this section we discuss some existing lexicons and annotated corpora for MWE identification.

[1] Translations obtained using Google's online translation service (http://translate.google.com) on September 6, 2017.

More$_B$ **often**$_I$ **than**$_I$ **not**$_I{}^{\text{MW-adverbial}}$,$_O$ however$_O$,$_O$ it$_O$ is$_O$ not$_O$ so$_O$ straightforward$_O$ to$_O$ **figure**$_B$ **out**$_I{}^{\text{VPC}}$ how$_O$ to$_O$ **make**$_B$ segmentation$_o$ **decisions**$_I{}^{\text{LVC}}$,$_O$ **in**$_B$ **order**$_I$ **to**$_I{}^{\text{MW-prep}}$ split$_O$ sentences$_O$ into$_O$ **lexical**$_B$ **units**$_I{}^{\text{MW-term}}$ that$_O$ **make**$_B$ **sense**$_I{}^{\text{idiom}}$

Fig. 1. Example of a sentence with MWEs identified (in bold), marked with BIO tags (subscripts) and disambiguated for their categories (superscripts). Source: [26].

Lexicons The simplest configuration of MWE identification requires only a list of entries that are to be treated as single tokens. Many parsers contain such lexicons, especially covering fixed MWEs such as compound conjunctions (e.g. *as well as*, *so that*) and prepositions *in spite of*, *up to*). Lists of MWEs with associated information can be found on language catalogues such as LDC and ELRA, but are also freely available, for instance, on the website of the SIGLEX-MWE section.[2] When the target constructions allow some morphological and/or syntactic variation, though, more sophisticated entry representations are required. Among the information given in MWE lexicons one usually founds the lemmas of the component words. This allows identifying MWE occurrences in inflected forms, if the text is lemmatised before identification. A complete survey of lexical resources containing MWEs is out of the scope of this work. For further reading on this topic, we recommend the excellent survey by [19].

Annotated corpora Identification of MWEs in running text can be modelled as a machine learning problem that learns from MWE-annotated corpora and treebanks. Many existing treebanks include some MWE annotations, generally focusing on a limited set of categories, as discussed in the survey by [29]. However, treebanks are not required for annotating MWEs in context. Minimally, tags can be used to delimit MWE occurrences. Additional tags or features can be used to classify MWE categories, as shown in Figure 1. Shared tasks often release free corpora for MWE identification. For instance, the SEMEVAL DIMSUM shared task focused on MWE identification in running text, releasing corpora with comprehensive MWE annotation for English [35].[3] The PARSEME shared task on verbal MWE identification released MWE-annotated corpora for 18 languages, focusing on verbal expressions only [32].[4] Other examples of annotated corpora with MWE tags include the English Wiki50 corpus [44], the English STREUSLE corpus [36], and the Italian MWE-anntoated corpus [40]. Some datasets focus on specific MWE categories, such as verb-object pairs [41] and verb-particle constructions [42, 1]. More rare but extremely relevant for MWE-aware MT, freely available parallel corpora annotated with MWEs also exist [45, 20, 25].

3 Rule-based MWE Identification

In rule-based identification, generally a lexicon is used to indicate which MWEs should be annotated in the text. In the simplest case, the lexicon contains only unambiguous fixed expressions that do not vary in inflection and in word order (e.g. *in fact, more often than not, even though*). In this case, a greedy string

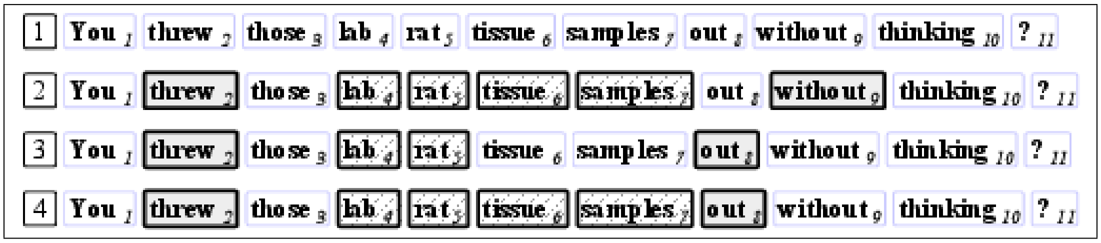
[2] http://multiword.sf.net/

[3] http://dimsum16.github.io

[4] http://multiword.sf.net/sharedtask2017

search algorithm suffices to match the MWE entries with the sentences. Special care must be taken if the target expressions are ambiguous, such as the fixed adverbial *by the way*, whose words can co-occur by chance as in *I recognise her by the way she walks* [8, 22]. Ambiguous fixed expressions, that can have compositional readings and/or accidental co-occurrence, require more sophisticated identification methods (e.g. the one described in Sec. 4).

Among semi-fixed unambiguous expressions that present only morphological inflection, nominal compounds such as *ivory tower* and *red herring* are frequent in many languages. The identification of this type of MWE is possible if the lexicon contains lemmatised entries, and if the text is automatically lemmatised prior to identification [24, 16]. Another alternative is to represent morphological inflection paradigms and restrictions in the lexicon, so that all alternative forms can be searched for when scanning the text [7, 31, 39].



1 You$_1$ threw$_2$ those$_3$ lab$_4$ rat$_5$ tissue$_6$ samples$_7$ out$_8$ without$_9$ thinking$_{10}$?$_{11}$

2 You$_1$ threw$_2$ those$_3$ lab$_4$ rat$_5$ tissue$_6$ samples$_7$ out$_8$ without$_9$ thinking$_{10}$?$_{11}$

3 You$_1$ threw$_2$ those$_3$ lab$_4$ rat$_5$ tissue$_6$ samples$_7$ out$_8$ without$_9$ thinking$_{10}$?$_{11}$

4 You$_1$ threw$_2$ those$_3$ lab$_4$ rat$_5$ tissue$_6$ samples$_7$ out$_8$ without$_9$ thinking$_{10}$?$_{11}$

Fig. 2. Lexicon-based MWE identification with the `mwetoolkit` using different match distances. Source: [12].

We have developed and evaluated several strategies for rule-based MWE identification, depending on the language, available resources and MWE categories. The following subsections summarise these methods, whose details can be found in previous publications [12, 13].

3.1 Lexicon-based matching

In [12], we propose a lexicon-based identification tool, developed as part of the mwetoolkit [24].[5] It was inspired on `jMWE` [15], a Java library that can be used to identify MWEs in running text based on preexisting MWE lists.

Proposed method The proposed software module allows more flexible matching procedures than `jMWE`, as described below. Moreover, the construction of MWE lists can be greatly simplified by using the MWE extractor integrated in the `mwetoolkit`. For example, given a noun compound pattern such as `Noun Noun`$^+$ and a POS-tagged corpus, the extractor lists all occurrences of this expression in a large corpus, which can in turn be (manually or automatically) filtered and passed on to the MWE identification module.

We propose an extension to the `mwetoolkit` which annotates input corpora based on either a list of MWE candidates or a list of patterns. In order to overcome the limitation of `jMWE`, our annotator has additional features described below.

1. **Different gapping possibilities**
   - Contiguous: Matches contiguous sequences of words from a list of MWEs.
   - Gappy: Matches words with up to a limit number of gaps in between.
2. **Different match distances**

[5] http://mwetoolkit.sf.net/

   - Shortest: Matches the shortest possible candidate (e.g. for phrasal verbs, we want to find only the closest particle).
   - Longest: Matches the longest possible candidate (e.g. for noun compounds).
   - All: Matches all possible candidates (useful as a fallback when shortest and longest are too strict).
3. **Different match modes**
   - Non-overlapping: Matches at most one MWE per word in the corpus.
   - Overlapping: Allows words to be part of more than one MWE (e.g. to find MWEs inside the gap of another MWE).
4. **Source-based annotation**: MWEs are extracted with detailed source information, which can later be used for quick annotation of the original corpus.

Examples Consider two different MWE patterns described by the POS regular expressions below:[6]

- `NounCompound` → `Noun Noun`$^+$
- `PhrasalVerb` → `Verb (Word`*`) Particle`

Given an input such as Sentence 1 (Figure 2) the gappy approach with different match distances will detect different types of MWEs. In Sentence 2, we show the result of identification using the *longest* match distance, which although well suited to identify noun compounds, may be too permissive for phrasal verbs combining with the closest particle (*out*). For the latter the *shortest* match distance will yield the correct response, but will be excessively strict when looking for a pattern such as the one for noun compounds, as shown in Sentence 3.

Discussion The proposed lexicon-based MWE identification module combines powerful generic patterns with a token-based identification algorithm with different matching possibilities. A wise choice of the best match distance is necessary when looking for patterns in corpora, and these new customisation possibilities allow identification under the appropriate conditions, so that one can achieve the result shown in Sentence 4 of Figure 2. With this module, one can either annotate a corpus based on a preexisting lexicon of MWEs or perform MWE type-based extraction, generate a lexicon and subsequently use it to annotate a corpus. When annotating the same corpus from which MWE types were extracted, source-based annotation can be used for best results.

One limitation of this approach concerns the occurrence of ambiguous expressions. Accidental co-occurrences would require contextual rules that might be tricky to express, and probably a context-dependent module would perform better for this kind of expression [22]. Moreover, since the module does not perform semantic disambiguation, an expression such as *piece of cake* would be annotated as an MWE in both sentences below:

1. *The test was a piece of cake*
2. *I ate a piece of cake at the bakery*

3.2 Corpus-based matching

While the proposal above has been tested only using preexisting MWE lexicons, we have subsequently employed it in a system submitted to the DiMSUM shared task and described in [13]. In this shared task, the competing systems were expected to perform both semantic tagging and MWE identification [35]. A

[6] In this toy example, the “lexicon” is formed by abstract POS patterns. In our implementation, lexicons can contain lemmas, surface forms, POS patterns or a mix of all these.

training corpus was provided containing annotated MWEs, both continuous and discontinuous (or gappy). The evaluation was performed on a test corpus provided to participants without any MWE annotation.

For MWE identification, we used a task-specific instantiation of the `mwetoolkit`, handling both contiguous and non-contiguous MWEs with some degree of customization, using the mechanisms described above. However, instead of using preexisting MWE lexicons, our MWE lexicons were automatically extracted from the training corpus, without losing track of their token-level occurrences. Therefore, we could guarantee that all the MWE occurrences learned from the training data were projected onto the test corpus.

Proposed method Our MWE identification algorithm uses 6 different rule configurations, targeting different MWE categories. While 3 of them are based on lexicons extracted from the training corpus, while the other 3 are unsupervised. The parameters of each configuration are optimized on a held-out development set, consisting of $\frac{1}{9}$ of the training corpus. The final system is the union of all configurations.

For the 3 supervised configurations, annotated MWEs are extracted from the training data and then filtered: we only keep combinations that have been annotated often enough in the training corpus. In other words, we keep MWE candidates whose proportion of annotated instances with respect to all occurrences in the training corpus is above a threshold t, discarding the rest. The thresholds were manually chosen based on what seemed to yield better results on the development set. Finally, we project the resulting MWE lexicons on the test data, that is, we segment as MWEs the test-corpus token sequences that are contained in the lexicon extracted from the training data. These configurations are:

- Contig: Contiguous MWEs annotated in the training corpus are extracted and filtered with a threshold of $t = 40\%$. That is, we create a lexicon containing all contiguous lemma+POS sequences for which at least 40% of the occurrences in the training corpus were annotated. The resulting lexicon is projected on the test corpus whenever that contiguous sequence of words is seen.
- Gappy: Non-contiguous MWEs are extracted from the training corpus and filtered with a threshold of $t = 70\%$. The resulting MWEs are projected on the test corpus using the following rule: an MWE is deemed to occur if its component words appear sequentially with at most a total of 3 gap words in between them.
- Noun^2-kn: We collect all noun-noun sequences in the test corpus that also appear at least once in the training corpus (known compounds), and filter them with a threshold of $t = 70\%$. The resulting list is projected onto the test corpus.

Additionally, we used 3 configurations based on POS patterns observed only on the test corpus. without looking at the training corpus.

- Noun^2-ukn: Collect all noun-noun sequences in the test corpus that never appear in the training corpus (unknown compounds), and project all of them back on the test corpus.
- Propn$^{2..\infty}$: Collect sequences of two or more contiguous words with POS-tag `PROPN` and project all of them back onto the test corpus.
- VP: Collect verb-particle candidates and project them back onto the test corpus. A verb-particle candidate is a pair of words under these constraints: the first word must have POS-tag `VERB` and cannot have lemma *go* or *be*. The two words may be separated by a `N`[7] or `PROPN`. The second word must be in a list of frequent non-literal particles[8]. Finally, the particle must be followed by a word with one of these POS-tags: `ADV`, `ADP`, `PART`, `CONJ`, `PUNCT`. Even though we might miss some cases, this final delimiter avoids capturing regular verb-PP sequences.

[7] In the remainder of the paper, we abbreviate the POS tag `NOUN` as `N`.

[8] The 13 most frequent non-literal particles: *about*, *around*, *away*, *back*, *down*, *in*, *into*, *off*, *on*, *out*, *over*, *through*, *up*.

Examples We have analysed some of the annotations made by the system and we show a sample of this analysis below:

- `N_N` Since our system looks for all occurrences of adjacent noun-noun pairs, we obtain a high recall for them. In 19 cases, however, our system has identified two `N`s that are not in the same phrase; e.g. **when I have a problem customer services don't want to know*. In order to realize that these nouns are not related, we would need parsing information. 17 cases have been missed due to only the first two nouns in the MWE being identified; e.g. **Try the memory foam pillows!* – instead of *memory foam pillows*. A similar problem occurred for sequences including adjectives, such as **My sweet pea plants arrived 00th May* – instead of *sweet pea plants*. In 24 cases, our system identified a compositional compound; e.g. **Quality gear guys, excellent!* Semantic features would be required to filter such cases out.
- **`VERB`-particles** Most of the `VERB_ADP` expressions were caught by the VP configuration, but we still had some false negatives. In 7 cases, the underlying particle was not in our list (e.g. *I regret ever going near their store*), while in 9 other cases, the particle was followed by a noun phrase (e.g. *Givin out Back shots*). 5 of the missed MWEs could have been found by accepting the particle to be followed by a `SCONJ`, or to be followed by the end of the line as delimiters. Most of the false positives were due to the verb being followed by an indirect object or prepositional phrase. We believe that disambiguating these cases would require valency information. 4 false positives were CONTIG cases of go to being identified as a MWE (e.g. **In my mother's day, she didn't go to college*). In the training corpus, this MWE had been annotated 57% of the time, but in future constructions (e.g. *Definitely not going to purchase a car from here*). Canonical forms would be easy to model with a specific contextual rule of the form *going to* ***`verb`***.

Discussion In spite of its simplicity, among the 9 submitted systems, our method was ranked 2nd in the overall results of the shared task. Three systems were ranked first, with two of them being submitted in the open condition (i.e. using external resources such as handcrafted lexicons).

In addition to simplicity, the system is also quite precise. Coverage is limited, though, to MWEs observed in the training corpus. Another limitation is that high-quality lemma and POS annotations are necessary to be able to extract reliable MWE lists from the training corpus and projecting them correctly on the test corpus. The manual tuning of rules and thresholds on a development set is effective, but also corpus-specific. Statistical methods like the ones described in Sec. 4 can be used to bypass this manual tuning step and build more general identification models.

4 Taggers for MWE Identification

A popular alternative, especially for contiguous semi-fixed MWEs, is to use an identification model that replaces the MWE lexicon. This model is usually learned using machine learning from corpora in which the MWEs in the sentences were manually annotated.

Machine learning techniques usually model MWE identification as a tagging problem based on BIO encoding,[9] as shown in Figure 1. In this case, supervised sequence learning techniques, such as conditional random fields [10] or a structured perceptron algorithm [34], can be used to build a model. It is also possible to combine POS tagging and MWE identification by concatenating MWE BIO and part-of-speech tags, learning a single model for both tasks jointly [11, 18].

We have developed and evaluated a statistical tagger for MWE identification based on conditional random fields. The following subsection summarise this methods, whose details can be found in a previous publication [37].

[9] B is used for a token that appears at the Beginning of an MWE, I is used for a token Included in the MWE, and O for tokens Outside any MWE.

4.1 CRF-based MWE identification

Linear-chain conditional random fields (CRFs) are an instance of stochastic models that can be used for sequence tagging [17]. Each input sequence T is composed of $t_1 \dots t_n$ tokens considered as an observation. Each observation is tagged with a sequence $Y = y_1 \dots y_n$ of tags corresponding to the values of the hidden states that generated them. CRFs can be seen as a discriminant version of hidden Markov models, since they model the conditional probability $P(Y|T)$. This makes them particularly appealing since it is straightforward to add customised features to the model. In linear-chain CRFs, the probability of a given output tag y_i for an input word x_i depends on the tag of the neighbour token y_{i-1}, and on a rich set of features of the input $\phi(T)$, that can range over any position of the input sequence, including but not limited to the current token t_i. CRF training consists in estimating individual parameters proportional to $p(y_i, y_{i-1}, \phi(T))$.

Proposed model The identification of continuous MWEs is a segmentation problem. In order to use a tagger to perform this segmentation, we use the well-known Begin-Inside-Outside (BIO) encoding [27]. In a BIO representation, every token t_i in the training corpus is annotated with a corresponding tag y_i with values `B`, `I` or `O`. If the tag is `B`, it means the token is the beginning of an MWE. If it is `I`, this means the token is inside an MWE. `I` tags can only be preceded by another `I` tag or by a `B`. Finally, if the token's tag is `O`, this means the token is outside the expression, and does not belong to any MWE. An example of such encoding for the 2-word expression *de la* (*some*) in French is shown in Figure 3.

i	-2	-1	0	1	2
w_i	*Il*	*jette*	*de*	*la*	*nourriture*
y_i	O	O	B	I	O
	He	*discards*	*some*		*food*

Fig. 3. Example of BIO tagging of a French sentence containing a *de*+determiner MWE, assuming that the current word (w_0) is *de*. Adapted from [37].

For our experiments, we have trained a CRF tagger using CRFSuite[10] [23]. We additionally allow the inclusion of features from external lexicons, such as the valence dictionary DicoValence[11] [14], and an automatically constructed lexicon of nominal MWEs obtained from the frWaC corpus [3] using the mwetoolkit [24]. Our features $\phi(T)$ contains 37 different combinations of values, inspired on those proposed by [10]:

- Single-token features (t_i):[12]
  - w_0 : wordform of the current token.
  - l_0 : lemma of the current token.
  - p_0 : POS tag of the current token.
  - w_i, l_i and p_i: wordform, lemma or POS of previous ($i \in \{-1, -2\}$) or next ($i \in \{+1, +2\}$) tokens.
- N-gram features (bigrams $t_{i-1}t_i$ and trigrams $t_{i-1}t_it_{i+1}$):
  - $w_{i-1}w_i$, $l_{i-1}l_i$, $p_{i-1}p_i$: wordform, lemma and POS bigrams of previous-current ($i = 0$) and current-next ($i = 1$) tokens.

[10] http://www.chokkan.org/software/crfsuite/

[11] http://bach.arts.kuleuven.be/dicovalence/

[12] t_i is a shortcut denoting the group of features w_i, l_i and p_i for a token t_i. In other words, each token t_i is a tuple (w_i,l_i,p_i). The same applies to n-grams.

  - $w_{i-1}w_iw_{i+1}$,$l_{i-1}l_il_{i+1}$, $p_{i-1}p_ip_{i+1}$: wordform, lemma and POS trigrams of previous-previous-current ($i = -1$), previous-current-next ($i = 0$) and current-next-next ($i = +1$) tokens.
- Orthographic features (ORTH):
  - hyphen and digits: the current wordform w_i contains a hyphen or digits.
  - f-capital: the first letter of the current wordform w_i is uppercase.
  - a-capital: all letters of the current wordform w_i are uppercase.
  - b-capital: the first letter of the current word w_i is uppercase, and it is at the beginning of a sentence ($i = 0$).
- Lexicon features (LF): These features depend on the provided lexicon and constitute either categorical labels or quantised numerical scores associated to given lemmas or lemma sequences.

Examples The CRF model described above was tested on French data, based on the French Treebank and on the French PARSEME shared task corpus. Experimental results can be found in [37]. Here, we present some examples of expressions identified and missed by the CRF tagger in the PARSEME shared task corpus.

In our error analysis, we wondered whether the CRF could predict MWEs that were never encountered in the training corpus. In the PARSEME test corpus, for instance, we can find the idiomatic expression *La musique n'adoucit pas toujours les moeurs* (*Music does not always soften the mores*). This expression was never seen in the training corpus and contains discontinuous elements, so the CRF could not identify it at all. Another interesting case is the continuous expression *remettre la main à la pâte* (lit. *to-put-again the hand in the dough*). Even though similar expressions occurred in the training test, such as *mettre la dernière main* (lit. *to-put the last hand*), this was not sufficient to identify the expression in the test set. In short, the CRF cannot locate expressions that were never seen in the training corpus, except if additional external lexicons are provided (which was not the case in this experiment).

Inversion of elements can also be problematic to identify for the CRF. For example, the sentence *une réflexion commune est menée* (lit. *a common reflection is lead*), contains an occurrence of the light-verb construction *mener réflexion* in passive voice. In the training corpus, we only see this expression in the canonical order, in active voice. Therefore, the CRF was not able to identify the expression, even though a variant had been observed in the training corpus.

Discussion This model can deal with ambiguous constructions more efficiently than rule-based ones, since it stores contextual information in the form of n-gram features. Moreover, there is no need to set thresholds, as these are implicitly modelled in the stochastic model. The discussion above underlines some of the limitations of the model: limited generalisation for constructions that have never been seen, and limited flexibility with respect to word order and discontinuities.

These limitations can be overcome using several techniques. The limited amount of training examples can be compensated with the use of external lexicons [10, 34, 28]. Discontinuities can be taken into account to some extent using more sophisticated encoding schemes [34], but the use of parsing-based MWE identification methods seems like a more appropriate solution [9]. Finally, better generalisation could be obtained with the use of vector representations for tokens, probably with the help of recurrent neural networks able to identify constructions that are similar to the ones observed in the training data, even though they do not contain the same lexemes.

5 Challenges in MWE Translation

We have presented three examples of systems performing monolingual MWE identification. Significant progress had been made in this field, including the construction and release of dedicated resources in many languages and the organisation of shared tasks. Current MWE identification systems could be used to detect expressions in the source text prior to translation. However, as we have seen in this paper, identification is not a solved problem, so care must be taken not to put the cart before the horses.

As noted by [8], MWE identification and translation share some challenges. First, discontinuities are a problem for both identification and translation. Continuous expressions can be properly dealt with by sequence models, both for identification and translation. However, many categories of expressions are discontinuous (e.g. verbal MWEs, as the ones in the PARSEME shared task corpora). Structural methods based on trees and graphs, both for identification and translation, are promising solutions that require further research.

Additionally, ambiguity is also a problem. For instance, suppose that an MT system learns that the translation of the English complex preposition *up to* into a foreign language is something that roughly corresponds to *until*. Then, the sentence *she looked it up to avoid confusion* would be incorrect and misleading. Context-aware systems such as the CRF described in Sec. 4 could be used to tag instances of the expression prior to translation. However, current MWE identification strategies for MT seem to be mostly rule-based [25, 4, 6, 7].

Identifying MWEs prior to translation is only part of the problem. Finding an appropriate translation requires access to parallel corpora instances containing the expression, external bilingual MWE lexicons and/or source-language semantic lexicons containing paraphrases and/or synonyms. Therefore, methods to automatically discover such resources could be employed as a promising solution to the MWE translation problem.

A final challenge concerns the evaluation of MWE translation. Many things can go wrong during MT, and MWEs are just one potential source of problems. Therefore, it is important to assess to what extent the MWE in a sentence was correctly translated. Dedicated manual evaluation protocols and detailed error typologies can be used [25], but automatic measures of comparison could also be designed, such as the ones proposed for MWE-aware dependency parsing [8].

References

1. Baldwin, T.: Deep Lexical Acquisition of Verb-particle Constructions. Computer Speech and Language 19(4), 398–414 (October 2005), http://dx.doi.org/10.1016/j.csl.2005.02.004
2. Baldwin, T., Kim, S.N.: Multiword expressions. In: Indurkhya, N., Damerau, F.J. (eds.) Handbook of Natural Language Processing, pp. 267–292. CRC Press, Taylor and Francis Group, Boca Raton, FL, USA, 2 edn. (2010)
3. Baroni, M., Bernardini, S. (eds.): Wacky! Working papers on the Web as Corpus. GEDIT, Bologna, Italy (2006), 224 p.
4. Barreiro, A., Monti, J., Batista, F., Orliac, B.: When multiwords go bad in machine translation. In: Mitkov, R., Monti, J., Pastor, G.C., Seretan, V. (eds.) Proceedings of the MT Summit 2013 Workshop on Multi-word Units in Machine Translation and Translation Technology (MUMTTT 2013). pp. 26–33. Nice, France (September 2013)
5. Calzolari, N., Fillmore, C., Grishman, R., Ide, N., Lenci, A., MacLeod, C., Zampolli, A.: Towards best practice for multiword expressions in computational lexicons. In: Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002). pp. 1934–1940. Las Palmas (2002)
6. Cap, F., Nirmal, M., Weller, M., im Walde, S.S.: How to account for idiomatic German support verb constructions in statistical machine translation. In: Proceedings of the 11th Workshop on Multiword Expressions (MWE 2015). pp. 19–28. Association for Computational Linguistics, Denver, Colorado, USA (2015), http://aclweb.org/anthology/W15-0903
7. Carpuat, M., Diab, M.: Task-based evaluation of multiword expressions: a pilot study in statistical machine translation. In: Proceedings of Human Language Technology: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2003). pp. 242–245. Association for Computational Linguistics, Los Angeles, California (June 2010), http://www.aclweb.org/anthology/N10-1029

8. Constant, M., Eryiğit, G., Monti, J., van der Plas, L., Ramisch, C., Rosner, M., Todirascu, A.: Multiword expression processing: A survey. Computational Linguistics (2017)
9. Constant, M., Nivre, J.: A transition-based system for joint lexical and syntactic analysis. In: Proc. of ACL 2016. pp. 161–171. Berlin, Germany (2016)
10. Constant, M., Sigogne, A.: MWU-aware part-of-speech tagging with a CRF model and lexical resources. In: Proc. of the ACL 2011 Workshop on MWEs. pp. 49–56. Portland, OR, USA (2011)
11. Constant, M., Tellier, I.: Evaluating the Impact of External Lexical Resources into a CRF-based Multiword Segmenter and Part-of-Speech Tagger. In: Proc. of LREC 2012. Istanbul, Turkey (2012)
12. Cordeiro, S., Ramisch, C., Villavicencio, A.: Token-based mwe identification strategies in the mwetoolkit. In: Proceedings of the 4th PARSEME General Meeting. Valetta, Malta (March 2015), https://typo.uni-konstanz.de/parseme/images/Meeting/2015-03-19-Malta-meeting/WG2-WG3-CORDEIRO-et-al-abstract.pdf
13. Cordeiro, S., Ramisch, C., Villavicencio, A.: Ufrgs&lif at semeval-2016 task 10: Rule-based mwe identification and predominant-supersense tagging. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016). pp. 910–917. Association for Computational Linguistics, San Diego, California (June 2016), http://www.aclweb.org/anthology/S16-1140
14. van den Eynde, K., Mertens, P.: La valence: l'approche pronominale et son application au lexique verbal. Journal of French Language Studies (13), 63–104 (2003)
15. Finlayson, M., Kulkarni, N.: Detecting multi-word expressions improves word sense disambiguation. In: Kordoni, V., Ramisch, C., Villavicencio, A. (eds.) Proceedings of the ALC Workshop on Multiword Expressions: from Parsing and Generation to the Real World (MWE 2011). pp. 20–24. Association for Computational Linguistics, Portland, OR, USA (June 2011), http://www.aclweb.org/anthology/W/W11/W11-0805
16. Kulkarni, N., Finlayson, M.: jMWE: A Java toolkit for detecting multi-word expressions. In: Kordoni, V., Ramisch, C., Villavicencio, A. (eds.) Proceedings of the ALC Workshop on Multiword Expressions: from Parsing and Generation to the Real World (MWE 2011). pp. 122–124. Association for Computational Linguistics, Portland, OR, USA (June 2011), http://www.aclweb.org/anthology/W/W11/W11-0818
17. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001). pp. 282–289. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001), http://dl.acm.org/citation.cfm?id=645530.655813
18. Le Roux, J., Rozenknop, A., Constant, M.: Syntactic parsing and compound recognition via dual decomposition: Application to French. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. pp. 1875–1885. Dublin City University and Association for Computational Linguistics, Dublin, Ireland (August 2014), http://www.aclweb.org/anthology/C14-1177
19. Losnegaard, G.S., Sangati, F., Escartín, C.P., Savary, A., Bargmann, S., Monti, J.: Parseme survey on MWE resources. In: Proceedings of LREC 2016. Portorož, Slovenia (2016)
20. Monti, J., Sangati, F., Arcan, M.: TED-MWE: a bilingual parallel corpus with mwe annotation: Towards a methodology for annotating mwes in parallel multilingual corpora. In: Proceedings of the Second Italian Conference on Computational Linguistics (CLiC-it 2015). Torino: Accademia University Press, Trento, Italy (2015)
21. Monti, J., Seretan, V., Pastor, G.C., Mitkov, R.: Multiword units in machine translation and translation technology. In: Mitkov, R., Monti, J., Pastor, G.C., Seretan, V. (eds.) Multiword Units in Machine Translation and Translation Technology. John Benjamin (2017)
22. Nasr, A., Ramisch, C., Deulofeu, J., Valli, A.: Joint dependency parsing and multiword expression tokenization. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 1116–1126. Association for Computational Linguistics, Beijing, China (July 2015), http://aclweb.org/anthology/P15-1108
23. Okazaki, N.: CRFsuite: a fast implementation of conditional random fields (CRFs) (2007), http://www.chokkan.org/software/crfsuite/
24. Ramisch, C.: Multiword Expressions Acquisition: A Generic and Open Framework, Theory and Applications of Natural Language Processing, vol. XIV. Springer (2015), http://link.springer.com/book/10.1007\%2F978-3-319-09207-2

25. Ramisch, C., Besacier, L., Kobzar, O.: How hard is it to automatically translate phrasal verbs from English to French? In: Mitkov, R., Monti, J., Pastor, G.C., Seretan, V. (eds.) Proceedings of the MT Summit 2013 Workshop on Multi-word Units in Machine Translation and Translation Technology (MUMTTT 2013). pp. 53–61. Nice, France (September 2013)
26. Ramisch, C., Villavicencio, A.: Computational treatment of multiword expressions. In: Mitkov, R. (ed.) Oxford Handbook of Computational Linguistics. Oxford University Press, 2nd edn. (2016)
27. Ramshaw, L., Marcus, M.: Text chunking using transformation-based learning. In: Third Workshop on Very Large Corpora (1995), http://aclweb.org/anthology/W95-0107
28. Riedl, M., Biemann, C.: Impact of mwe resources on multiword recognition. In: Proceedings of the 12th Workshop on Multiword Expressions (MWE 2016). pp. 107–111. Association for Computational Linguistics, Berlin, Germany (2016), http://anthology.aclweb.org/W16-1816
29. Rosén, V., De Smedt, K., Losnegaard, G.S., Bejcek, E., Savary, A., Osenova, P.: MWEs in treebanks: From survey to guidelines. In: Proceedings of LREC 2016. pp. 2323–2330. Portorož, Slovenia (2016)
30. Sag, I., Baldwin, T., Bond, F., Copestake, A., Flickinger, D.: Multiword expressions: A pain in the neck for NLP. In: Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2002). Lecture Notes in Computer Science, vol. 2276/2010, pp. 1–15. Springer, Mexico City, Mexico (February 2002)
31. Savary, A.: Multiflex: A multilingual finite-state tool for multi-word units. In: CIAA. vol. 5642, pp. 237–240 (2009), http://dx.doi.org/10.1007/978-3-642-02979-0_27
32. Savary, A., Ramisch, C., Cordeiro, S., Sangati, F., Vincze, V., QasemiZadeh, B., Candito, M., Cap, F., Giouli, V., Stoyanova, I., Doucet, A.: The parseme shared task on automatic identification of verbal multiword expressions. In: Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017). pp. 31–47. Association for Computational Linguistics, Valencia, Spain (2017), http://aclweb.org/anthology/W17-17
33. Savary, A., Sailer, M., Parmentier, Y., Rosner, M., Rosén, V., Przepiórkowski, A., Krstev, C., Vincze, V., Wójtowicz, B., Losnegaard, G.S., Parra Escartín, C., Waszczuk, J., Constant, M., Osenova, P., Sangati, F.: PARSEME – parsing and multiword expressions within a European multilingual network. In: Proc. of LTC 2015. Poznań (2015)
34. Schneider, N., Danchik, E., Dyer, C., Smith, N.A.: Discriminative lexical semantic segmentation with gaps: running the MWE gamut. TACL 2, 193–206 (2014)
35. Schneider, N., Hovy, D., Johannsen, A., Carpuat, M.: Semeval-2016 task 10: Detecting minimal semantic units and their meanings (diMSUM). In: Proc. of SemEval 2016. pp. 546–559. San Diego, CA, USA (2016)
36. Schneider, N., Onuffer, S., Kazour, N., Danchik, E., Mordowanec, M.T., Conrad, H., Smith, N.A.: Comprehensive annotation of multiword expressions in a social web corpus. In: Proceedings of LREC 2014. pp. 455–461. Reykjavik, Iceland (2014)
37. Scholivet, M., Ramisch, C.: Identification of ambiguous multiword expressions using sequence models and lexical resources. In: Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017). pp. 167–175. Association for Computational Linguistics, Valencia, Spain (2017), http://aclweb.org/anthology/W17-1723
38. Seretan, V.: On translating syntactically-flexible expressions. In: Mitkov, R., Monti, J., Pastor, G.C., Seretan, V. (eds.) Proceedings of the MT Summit 2013 Workshop on Multi-word Units in Machine Translation and Translation Technology (MUMTTT 2013). pp. 11–11. Nice, France (September 2013)
39. Silberztein, M.: The lexical analysis of natural languages. In: Finite-State Language Processing, pp. 175–203. MIT Press (1997)
40. Taslimipoor, S., Desantis, A., Cherchi, M., Mitkov, R., Monti, J.: Language resources for italian: towards the development of a corpus of annotated italian multiword expressions. In: Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016), Napoli, Italy, December 5-7, 2016. (2016)
41. Tu, Y., Roth, D.: Learning English light verb constructions: Contextual or statistical. In: Kordoni, V., Ramisch, C., Villavicencio, A. (eds.) Proceedings of the ALC Workshop on Multiword Expressions: from Parsing and Generation to the Real World (MWE 2011). pp. 31–39. Association for Computational Linguistics, Portland, OR, USA (June 2011), http://www.aclweb.org/anthology/W/W11/W11-0807

42. Tu, Y., Roth, D.: Sorting out the Most Confusing English Phrasal Verbs. In: Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. pp. 65–69. SemEval '12, Association for Computational Linguistics, Stroudsburg, PA, USA (2012)
43. Vauquois, B.: A survey of formal grammars and algorithms for recognition and transformation in mechanical translation. In: IFIP Congress (2). pp. 1114–1122 (1968)
44. Vincze, V., Nagy, I., Berend, G.: Multiword expressions and named entities in the Wiki50 corpus. In: Proceedings of RANLP 2011. pp. 289–295. Hissar, Bulgaria (2011)
45. Vincze, V.: Light verb constructions in the SzegedParalellFX English–Hungarian parallel corpus. In: Proceedings of LREC 2012. pp. 2381–2388. Istanbul, Turkey (2012)