Multi-Word Expression Identification Using Sentence Surface Features

Ram Boukobza
School of Computer Science
Hebrew University of Jerusalem
ram.boukobza@mail.huji.ac.il

Ari Rappoport
School of Computer Science
Hebrew University of Jerusalem
arir@cs.huji.ac.il

Abstract

Much NLP research on Multi-Word Expressions (MWEs) focuses on the discovery of new expressions, as opposed to the identification in texts of known expressions. However, MWE identification is not trivial because many expressions allow variation in form and differ in the range of variations they allow. We show that simple rule-based baselines do not perform identification satisfactorily, and present a supervised learning method for identification that uses sentence surface features based on expressions' canonical form. To evaluate the method, we have annotated 3350 sentences from the British National Corpus, containing potential uses of 24 verbal MWEs. The method achieves an F-score of 94.86%, compared with 80.70% for the leading rule-based baseline. Our method is easily applicable to any expression type. Experiments in previous research have been limited to the compositional/non-compositional distinction, while we also test on sentences in which the words comprising the MWE appear but not as an expression.

1 Introduction

Multi-Word Expressions (MWEs) such as 'pull strings', 'make a face' and 'get on one's nerves' are very common in language. Such MWEs can be characterized as being *non-compositional*: the meaning of the expression does not transparently follow from the meaning of the words that comprise it. Much of the work on MWEs in NLP has been in *MWE extraction* – the discovery of new MWEs from a corpus, using statistical and other methods. Identification of *known* MWEs in text has received less attention, but is necessary for many NLP applications, for example in machine translation. The current work deals with the *MWE identification* task: deciding if a sentence contains a use of a known expression.

MWE identification is not as simple as may initially appear, as will be shown by the performance of two rule-based baselines in our experiments. One source of difficulty is variations in expressions' usage in text. Although MWEs generally show less variation than single words, they show enough that it cannot be ignored. In a study on V+NP idioms, Riehemann (2001) found that the idioms' canonical form accounted for 75% of their appearances in a corpus. Additionally, expressions differ considerably in the types of variations they allow, which include passivization, nominalization and addition of modifying words (Moon, 1998).

A second source of difficulty is that expressions consisting of very frequent words will often co-occur in sentences in a non-MWE usage and in similar but distinct expressions.

MWE identification can be modeled as a two step process. Given a sentence and a known expression, step (1) is to decide if the sentence contains a potential use of the expression. This is a relatively simple step based on the appearance in the sentence of the words comprising the MWE. Step (2) is to decide if the potential use is indeed non-compositional. Consider the following sentences with regard to the expression *hit the road*, meaning 'to leave on a journey':

(a) 'At the time, *the road* was long and difficult with few travelers daring to take it.'

(b) 'The headlights of the taxi-van behind us flashed as it *hit* bumps in *the road*.'

(c) 'The bullets were *hitting the road* and I could see them coming towards me a lot faster than I was able to reverse.'

(d) 'Lorry trailers which would have been *hitting the road* tomorrow now stand idle.'

Sentence (a) does not contain a potential use of the expression due to the missing component 'hit'. Each of (b)-(d) does contain a potential use of the expression. In (b) all of the expression components are present, but they do not form an expression. In (c), the words form an expression, but with a compositional (literal) meaning. Only (d) contains a non-compositional use of *hit the road*. The task we address in this paper is to identify whether or not we are in case (d), for a given expression in a given sentence.

To date, most work in MWE identification has focused on manually encoding rules that identify expressions in text. The encodings, usually consisting of regular expressions and syntactic structures, are intended to contain all the necessary information for processing the MWE in text. Being manual, this is time-consuming work and requires expert knowledge of individual expressions. In terms of the above model, such encodings handle both MWE identification steps.

A second approach is to use machine learning methods to learn an expression's behavior from a corpus. Studies taking this approach have focused on distinguishing between compositional and non-compositional uses of an expression (cases (c) and (d) above). As will be detailed in Section 2, existing methods are tailored to an expression's type, and experiment with a single MWE pattern. In addition, the training and test sets they used did not contain non-expression uses as in case (b), which can be quite common in practice.

Our approach is more general. Given a set of sentences with potential MWE uses, we use sentence surface features to create a Support Vector Machine (SVM) classifier for each expression. The classifier is binary and differentiates between non-compositional uses of the expression ((d) above) on the one hand, and compositional and non-expression uses ((b) and (c)) on the other. The experiments and results presented below focus on verbal MWEs, since verbal MWEs are quite common in language use and have also been investigated in related MWE research (e.g., (Cook et al., 2007)). However, the developed features are not specific to a particular type of expression.

The supervised method is compared with two simple rule-based baselines in order to test whether a simple approach is sufficient. In addition, the use of surface features is compared with the use of syntactic features (based on dependency parse trees of the sentences). Averaged over expressions in an independent test set, the supervised classifiers outperform the rule-based baselines, with F-scores of 94.86% (surface features) and 87.77% (syntactic features), compared with 80.70% for the best baseline.

Section 2 reviews previous work. Section 3 discusses the features used for the supervised classifier. Section 4 explains the experimental setting. The results and a discussion are given in sections 5 and 6.

2 Previous Work

2.1 MWE Lexical Encoding

The approach to handling MWEs in early systems was to employ a list of expressions, each with a quasi regular expression that encodes morpho-syntactic variations. One example is Leech et al. (1994) who used this method for automatic part-of-speech tagging for the BNC. Another is a formalism called IDAREX (IDioms And Regular EXpressions) (Breidt et al., 1996).

More recent research emphasizes the integration of MWE lexical entries into existing single word lexicons and grammar systems (Villavicencio et al., 2004; Alegria et al., 2004). There is also an attempt to take advantage of regularities in morpho-syntactic properties across MWE groups, which allows encoding the behavior of the group instead of individual expressions (Villavicencio et al., 2004; Grégoire, 2007). Fellbaum (1998) discusses some difficulties in representing idioms, which are largely figurative in meaning, in WordNet. More recent work (Fellbaum et al., 2006) focuses on German VP idioms.

As already mentioned, one issue with lexical encoding is that it is done manually, making lexicons difficult to create, maintain and extend. The use of regularities among different types of MWEs is one way of reducing the amount of work required. A second issue is that implementations tend to ignore the likelihood and even the possibility of compositional and other interpretations of expressions in text, which can be common for some expressions. For example, in an MWE identification study, Hashimoto et al. (2006) built an identification system using hand crafted rules for some 100 Japanese idioms. The results showed near perfect performance on expressions without compositional/non-compositional ambiguity but significantly poorer performance on expressions with ambiguity.

2.2 MWE Identification by ML

Katz and Giesbrecht (2006) used a supervised learning method to distinguish between compositional and non-compositional uses of an expression (in German text) by using contextual information in the form of Latent Semantic Analysis (LSA) vectors. LSA vectors of compositional and non-compositional meaning were built from a training set of example sentences and then a nearest neighbor algorithm was applied on the LSA vector of one tested MWE. The technique was tested more thoroughly in Cook et al. (2007).

Cook et al. (2007) devised two unsupervised methods to distinguish between compositional (literal) and non-compositional (idiomatic) tokens of verb-object expressions. The first method is based on an expression's canonical form. In a previous study (Fazly and Stevenson, 2006), the authors came up with a dozen possible syntactic forms for verb-object pairs (based on passivization, determiner, and object pluralization) and used a corpus-based statistical measure to determine the canonical form(s). The method classifies new tokens as idiomatic if they use a canonical form, and literal otherwise.

The second method uses context as well as form. Co-occurrence vectors representing the idiomatic and literal meaning of each expression were computed based on corpus data. Idiomatic-meaning vectors were based on examples matching the expressions' canonical form. Literal meaning vectors were based on examples that did not match the canonical form. New tokens were classified as literal/idiomatic based on their (co-occurrence) vector's cosine similarity to the idiomatic and literal vectors.

(Sporleder and Li, 2009) also attempted to distinguish compositional from non-compositional uses of expressions in text. Their assumption was that if an expression is used literally, but not idiomatically, its component words will be related semantically to several words in the surrounding discourse. For example, when the expression 'play with fire' is used literally, words such as 'smoke, 'burn', 'fire department', and 'alarm' tend to also be used nearby; when it is used idiomatically, they aren't (indeed, other words, e.g., 'danger' or 'risk' appear nearby but they are not close semantically to 'play' or to 'fire'). This property was used to distinguish literal and non-literal instances by measuring the semantic relatedness of an expression's component words to nearby words in the text. If one or more of the expression's components were sufficiently related to enough nearby words, forming a 'lexical chain', the usage was classified as literal. Otherwise it was idiomatic. Two classifiers based on lexical chains were devised. These were compared with a supervised method that trains a classifier for each expression based on surrounding context. The results showed that the supervised classifier method did much better (90% F-score on literal uses) than the lexical chain classifier methods (60% F-score).

In the above studies the focus is on the compositional/non-compositional expression distinction. The sentence data used contains examples of either one or the other. In (Sporleder and Li, 2009) the experimental data included only sentences in which the expressions were in canonical form (allowing for verb inflection). In (Cook et al., 2007) a syntactic parser was used to collect sentences containing the MWEs in the active and passive voice using heuristics. Thus, examples such as the following (from the BNC) would not be included in their sample:

1. *take a chance*: 'While he still had **a chance** of being near Maisie, he would **take** it'.

2. *face the consequences*: '. . . she did not have to **face**, it appears, **the** possible serious or even fatal **consequences** of her decision'.

3. *make a distinction*: 'Logically, **the distinction** between the two aspects of the theory can and should be **made**'.

4. *break the ice*: '**The ice**, if not **broken**, was beginning to soften a little'.

5. *settle a score*: 'Morrissey had another **score to settle**'.

This means that their experiments have not included all types of sentences that might be encountered in practice when attempting MWE identification. Specifically, they would miss many examples in which the MWE words are present but are not used as an expression (case (b) in Section 1). Moreover, their heuristics are tailored to the Verb-Direct Object MWE type. Different heuristics would need to be employed for different MWE types.

In our approach there is no pre-processing stage requiring type-specific knowledge. Specifically, the above examples are used as training sentences in our experiments.

2.3 MWE Extraction

There exists an extensive body of research on MWE extraction (see Wermter and Hahn (2004) for a review), where the only input is a corpus, and the output is a list of MWEs found in it. Most methods collect MWE candidates from the corpus, score them according to some association measure between their components, and accept candidates with scores passing some threshold. The focus of research has been on developing association measures, including statistical, information-theoretic and linguistically motivated measures (e.g., Justeson and Katz (1995), Wermter and Hahn (2006), and Deane (2005)).

3 MWE Identification Method

Our method decides if a potential use of a known expression in a given sentence is non-compositional. The input to the method, for each MWE, is a labeled training set of sentences containing one or more potentially non-compositional uses of the MWE. The output, for each MWE, is a binary classifier, trained on those sentences. Thus, we target step (2) of MWE identification, which is the difficult one.

The learning algorithm used is Support Vector Machine (SVM), which outputs a binary classifier, using Sequential Minimal Optimization (Platt, 1998)[1] in the Weka toolkit[2] (Witten and Frank, 2000).

For training, sentences are converted into feature vectors. Features depend on the assignment of the lexical components of the expression to specific tokens in the sentence. In some cases, there are several tokens in the sentence that match a single component in the expression, and this leads to multiple (potential) assignments. So in the general case a sentence is converted to a set of feature vectors, each corresponding to a single assignment of the MWE's lexical components to sentence tokens.

Training sentences are labeled positive if they contain a non-compositional use of the expression and negative if they do not (i.e., literal and other uses). If the sentence is positive, at least one of the assignments is the true assignment (there may be more than one, e.g., when an expression is used twice in the same sentence). The vector matching the true assignment is labeled positive. The others are labeled negative. If the sentence is negative, all of the vectors are labeled negative.

As mentioned, the output of the method is a distinct binary classifier for each MWE. Although having a single classifier for all expressions would seem advantageous, the wide variation exhibited by MWEs (e.g., for some the passive is common, for other not at all) precludes this option and requires having a separate classifier for each expression.

3.1 Features

Surface features include order and distance, part-of-speech and inflection of an expression's words in a sentence.

Use of surface features is intuitive and relatively cheap. In addition, many studies have shown the importance of order and distance in MWE extraction in English (two recent examples are (Dias, 2003; Deane, 2005)). Thus, we develop a supervised classifier based on surface features.

Many of the surface features make use of an expression's Canonical Form (CF), thus the learning algorithm assumes that it is given such a form. Formally defining the CF is difficult. Indeed, some researchers have concluded that some expressions do not have a CF (Moon, 1998). For our purposes, CF can be informally defined as the most frequent form in which the expression appears. In practice, an approximation of this definition, explained in Section 4, is used.

3.1.1 Surface Features

1. Word Distance: The number of words between the leftmost and rightmost MWE tokens in the sentence.

2. Ordered Gap List: A list of gaps, measured in number of words, between each pair of the

[1] Using the PUK kernel (The Pearson VII function-based Universal Kernel), with parameters omega=1.0 and sigma=1.0.

[2] Weka version 3.5.6; www.cs.waikato.ac.nz/ ml/ weka/

expression's tokens in their canonical form order. For example, if the token locations (in canonical form order) are 10, 7 and 3, the ordered gap list would be $(10 \leftrightarrow 7 = 2, 10 \leftrightarrow 3 = 6, 7 \leftrightarrow 3 = 3)$.

3. Word Order: A boolean value indicating whether the expression's word order in the sentence matches the canonical form word order.

4. Word Order Permutation: The permutation of word order relative to the canonical form. For example, the permutation (1,0,2) indicates that component words 1 and 0 have switched order in the sentence.

5. Inflection Ratio: The fraction of words in the expression that have undergone inflection relative to the canonical form.

6. Lexical Values: A list of the tokens in the sentence matching the expression's component words, ordered according to canonical form. For example, if the expression is 'make a distinction', a possible lexical values list is (made,no,distinction) in the sentence 'No possible distinction can be made between the two'.

7. POS Pattern: A boolean value indicating whether the expression's use in the sentence has the same part-of-speech pattern as the canonical form.

Two combinations of surface features are used in the experiments below. The first, named R1, uses all of the above features. The second, R2, uses only Word Distance, Ordered Gap List and Word Order Permutation. Using R2 the learner has only word order and distance information from which to create a classifier.

3.1.2 Syntactic Features

An expression's words may appear unrelated in a sentence, because of distance, order, part-of-speech and other surface variations. However, the words will still be closely related syntactically. Syntactic analysis of the sentence in the form of a dependency parse tree directly gives the syntactic relationships between the expression's components. Thus, we also develop a classifier based on syntactic features.

Dependency Parsing. A dependency parse tree is a directed acyclic graph in which the nodes represent tokens in the sentence and the edges represent syntactic dependencies between the words (e.g., direct-object, prepositional-object, noun-subject etc.). The Stanford Parser[3] (Marneffe et al., 2006) was used.

Minimal Sub-Tree. To compute a syntactic feature, the dependency tree is computed and then the minimal sub-tree containing the expression's tokens is extracted.

The features are:

1. Sub-Tree Distance Sum: The number of edges in the minimal sub-tree. A large number of edges suggests a weaker dependency.

2. Sub-Tree Distance List: A list of the distances of the MWE component nodes from the root of their sub-tree.

3. Descendant Relations List: A list of descendant relations between each pair of MWE component nodes.
   A descendant relation between two nodes exists if there is a directed path from one node (the ancestor) to the other (the descendant). Descendant relations are either direct (parent-child) or indirect. The list consists of the levels of descendant relations between the MWE component nodes, which can be none, indirect or direct.

4. Descendant Direction List: A list of the directions of the descendant relations between each pair of MWE component nodes.
   If there are descendant relations between a pair of nodes, the direction of the dependency, indicating which is the modifying and which the modified node, is important.

5. Sibling Relations List: A list of sibling relations between each pair of MWE component nodes.
   Two nodes are first degree siblings if they share the same parent (which usually means they modify the same word). Two nodes are second degree siblings if they share a common ancestor no more than two edges away, and so on. The list consists of the level of sibling relations for each pair of component

[3] http://www-nlp.stanford.edu/software/lex-parser.shtml

nodes, which can be first, second and third degree.

6. Descendant Type List: A list of the dependency types (e.g., subject, direct object etc.) between each pair of component nodes. If the component nodes are not direct descendants their dependency type is null.

7. Sibling Type List: A list of pairs of dependency types corresponding to the dependencies between a pair of component nodes and their common parent. If the component nodes are not first degree siblings, the type is null.

In the experiments reported below, the classifier using only the syntactic features is denoted by S, and the one using all surface and all syntactic features is denoted by C. We have experimented with additional feature combinations, with no improvement in results.

4 Experimental Method

Canonical form. As described, an expression's canonical form (CF) is used in many of the learning algorithm's features. The CF is taken from Collins COBUILD Advanced Learner's English Dictionary (2003) which is also used as our source for MWEs. COBUILD is an English-English dictionary based on the Bank of English (BOE) corpus (over 520 million words) with approximately 34,000 entries.

Traditional single-word dictionaries are a good source for expressions because they usually list, as part of single-word entries, expressions in which the word is a component. The CF is not explicitly given in COBUILD, so an approximation is the form which appears in the expression's definition. This is a reasonable approximation since the COBUILD authors claim to have selected *typical* uses of the expressions in their definitions.

Each CF also has a matching part-of-speech (POS) pattern, which is a list of the parts-of-speech of the components in the CF. For example, 'walking on air' has the pattern $(Verb, Preposition, Noun)$. COBUILD does not include part-of-speech information for expressions so this information was determined using the British National Corpus (BNC) (BNC, 2001), a (mostly) automatically POS tagged corpus (using the CLAWS tagger). For each MWE, the POS patterns of all instances of the CF in the corpus were counted. The most frequent pattern is the expression's POS pattern.

The expressions. A set of 17 verbal MWEs, the development set, was used for development of the surface and syntactic features described above. All of the development set MWEs had the POS pattern $(Verb, Determiner, Noun)$. Another set of 24 verbal MWEs, the training/test set[4], was then used to test the method. Because the method is not specific to the $(Verb, Determiner, Noun)$ pattern, new POS patterns are included in the training/test set. The training/test set consists of 8 MWEs of the POS pattern $(Verb, Determiner, Noun)$, 7 $(Verb, Preposition, Noun)$ MWEs and and 9 $(Verb, Noun, Preposition)$ MWEs. The list of MWEs was selected randomly from the corresponding POS pattern types. MWEs with a positive or negative percentage of under 5% in their data set were discarded[5]. The MWEs, in their canonical form, are:
Development set:
$(Verb, Determiner, Noun)$ [17]: *break the ice, calls the shots, catch a cold, clear the air, face the consequences, fits the bill, hit the road, make a face, make a distinction, makes an impression, raise the alarm, set an example, sound the alarm, stay the course, take a chance, take the initiative, tie the knot.*
Training/test set:
$(Verb, Determiner, Noun)$ [8]: *changes the subject, get a grip, get the picture, lead the way, makes the grade, sets the scene, take a seat, take the plunge*;
$(Verb, Preposition, Noun)$ [7]: *fall into place, goes to extremes, brought to justice, take to heart, gets on nerves, keep up appearances, comes to light*;
$(Verb, Noun, Preposition)$ [9]: *take aim at, make allowances for, takes advantage of, keep hands off, lay claim to, take care of, make contact with, gives rise to, wash hands of.*

The sentences. As mentioned, the first step of MWE identification is to identify if the sentence contains a potential non-compositional use of the expression. In order to test our method, which targets step (2), a set of such sentences (for each expression) was collected from the BNC corpus and

[4] Using 10-fold cross validation.

[5] Initially there were 20 MWEs in the development set and 30 (10 per group) in the training/test set.

then labeled for use as training/test sentences[6].

The collection method was intended to allow a wide range of variations in expression use. In practice, for each expression sentences containing all of the expression's CF components, in any of their inflections, were collected, but excluding common auxiliary words. So for example, when targeting the MWE 'make an impression' we allowed inflections of 'make' and 'impression' and did not require 'an', to allow for variations such as 'make no impression' and 'make some impression'. For some expressions, sentences were limited to those with a distance of up to 8 words between each expression component. Very long sentences (above 80 words) were discarded. The final set of sentences was then randomly selected.

Given this method, training/test sentences allow non-lexical variations: inflection, word order, part-of-speech, syntactic structure and other non-syntactic transformations. Lexical variations which involve a change in one of the expression's components are not allowed, except for common auxiliary words.

For the development set an average of 97 (40-137) sentences were collected per MWE, giving a total of 1663 sentences, with a micro average of 49% positive labels. For the training/test set there were 139 (73-150) sentences per MWE on average, totaling 3350, with a 40% average positive ratio.

The sentences were manually labeled as positive if they contained a non-compositional use of the MWE and negative if they contained a compositional or non-expression usage. Judgment was based on a single sentence, without wider context.

Baseline methods. Two baseline methods are used to test the intuitive notion that simple rule-based methods are sufficient for MWE identification as well as for comparison with the supervised learning methods.

The first method, CanonicalForm (CF), accepts a sentence use as a non-compositional MWE use if and only if the MWE is in canonical form (there are no intervening words between the MWE components, their order matches canonical-form order, and there is an inflection in at most one component word).

The second method, DistanceOrder (DO), accepts a sentence use if and only if the number of words between the leftmost and rightmost MWE components is less than or equal to 2 (not counting the middle MWE component), and if the order matches the canonical form order.

	CF	DO	R1	R2	S	C
Verb-Det-Noun: All (17)						
A	73.53	82.27	89.48	**90.83**	88.58	87.02
P	**97.09**	89.29	82.71	87.18	83.89	78.54
R	58.81	76.83	92.29	90.35	92.97	**97.19**
F	67.39	79.68	86.92	**88.56**	87.78	86.00
Verb-Det-Noun: Best (8)						
A	84.51	91.56	95.33	**95.48**	92.52	93.27
P	**95.90**	85.70	92.50	95.63	91.12	87.63
R	73.50	89.80	97.25	95.25	95.83	**98.50**
F	78.63	86.29	94.70	**95.36**	93.44	92.25

Table 1: Development set: Average performance over all MWEs and best 8. Supervised classifiers outperform baselines. **A**: Accuracy; **P**: Positive Precision; **R**: Positive Recall; **F**: F-Score.

5 Results

The baseline methods (CF and DO) and the supervised methods (R1,R2,S,C) were run on the development and training/test sets. For the supervised methods, for each MWE we used 10-fold cross-validation[7].

Tables 1 and 2 summarize the results for the development and test sets, respectively. For the development set, average results over all 17 MWEs and over the best 8 MWEs (on R1), a group size comparable to the test set, are shown. For the test set, results over all 24 MWEs and the three MWE types tested are shown.

The tables show average overall accuracy and average precision, recall and F-score on positive instances, where the averages are taken over the results of the individual MWEs (i.e., micro-averaged).

Baselines. Baseline accuracy, (for DO) 82.27% on the development set and 87.2% on the test set (over all groups), is probably insufficient for many NLP applications.

The baselines perform similarly in terms of average accuracy. CF does this with very high precision and low recall, while for DO recall improves at the expense of precision. Looking at individual MWEs reveals that for expressions which allow more variation in terms of intervening words

[6] The PyLucene software package, http://pylucene. osafoundation. org/, was used for building an index to the BNC and for searching.

[7] I.e., we ran 10 experiments where in each experiment we divided the corresponding annotated sentence sets into 90% training sentences and 10% test sentences, and the results reported are the average of the 10 experiments.

	CF	DO	R1	R2	S	C
All (24)						
A	86.16	87.15	**93.50**	91.61	89.73	91.50
P	**94.16**	80.38	93.08	93.16	89.86	89.26
R	68.86	86.88	93.00	89.74	88.94	**93.33**
F	75.53	80.70	**94.86**	93.09	87.77	92.80
Verb-Det-Noun (8)						
A	89.08	89.08	**93.83**	93.65	90.07	91.33
P	**95.44**	84.13	92.88	94.00	91.04	89.25
R	73.30	88.53	97.50	95.50	91.57	**97.63**
F	80.97	84.91	**95.09**	94.71	91.21	93.08
Verb-Prep-Noun (7)						
A	85.53	91.15	**93.64**	92.62	88.75	92.10
P	97.13	81.40	96.81	**97.20**	92.48	94.33
R	64.36	**92.67**	84.73	82.79	82.71	85.00
F	74.08	86.03	**97.81**	96.87	83.13	96.65
Verb-Noun-Prep (9)						
A	84.06	82.32	**93.11**	88.99	90.18	91.18
P	90.72	76.26	**90.78**	89.73	86.78	85.89
R	68.41	80.90	95.44	90.03	91.44	**96.00**
F	71.82	72.82	**92.69**	89.14	88.33	89.99

Table 2: Test set: Average performance over all MWEs and by group. The best supervised classifier outperforms baselines in all groups. **A**: Accuracy; **P**: Precision; **R**: Recall; **F**: F-Score.

and lexical change, DO outperforms CF. To name a few, *make an impression, raise the alarm, take a chance* and *make allowances for*. For example, for *take a chance* intervening words are quite common, as in: 'I'm taking a real chance on you.', or a change in determiner as in: 'I preferred to take my chances'. Indeed, CF showed poor precision only for MWEs with a common literal usage. Two such MWEs were present in the development set (*break the ice* and *tie the knot*) and two in the test set (*wash hands of* and *keep hands off*).

Baselines versus supervised classifiers. As shown in the tables, R1 outperforms the best baseline in terms of accuracy in both test and development. Moreover, the supervised classifiers are more stable in their accuracy. For the development set the standard deviation of accuracy scores averages 22.58 for CF and DO, and 6.68 for R1, R2, S, and C. For the test set the baselines average 9.07 (Verb-Det-Noun), 11.11 (Verb-Prep-Noun) and 14.26 (Verb-Noun-Prep), and the supervised methods average 4.97 (Verb-Det-Noun), 7.66 (Verb-Prep-Noun) and 7.97. This stability means that the supervised classifiers are able to perform well on MWEs with different behavior. For example, R1 is able to perform well on expressions where order is strict, as DO does (e.g., *make a face*), while also performing well on those where order varies (e.g., *make a distinction*).

Supervised classifiers. R1 and R2, based on surface features, show similar accuracy values, with R1 doing somewhat better in the Verb-Prep-Noun and Verb-Noun-Prep groups. This is due to the Lexical Values feature, which accounts for a change in preposition. A change in preposition (as in 'wash hands *of* some matter' versus 'wash hands *in* the sink') is more significant than a change in determiner in the Verb-Determiner-Noun group. This improves precision on negative instances, which are rejected more precisely based on the preposition value. Nevertheless, the relatively simple features in R2, essentially order and distance, perform quite well.

The F-score result for R1, 94.86, is an improvement over the F-score result of the supervised classifier used in (Sporleder and Li, 2009), 90.15. Although the sentence data is different (our data includes sentences with non-expression uses) the number of sentences used is similar.

S, based on syntactic features, performs worse than R1/2. It shows better accuracy than the baselines in all but the (Verb-Prep-Noun) group and is also more stable. C, a combination of surface and syntactic features, performs better than S and slightly worse than R1/2.

Why do the syntactic features perform worse than surface features? An analysis of the S classifier errors reveals two important causes. First, there is substantial variation in the dependency tree structures of the non-compositional uses of the expressions as output by the parser. Thus, the syntactic feature classifier was more difficult to learn than the surface feature one, requiring a larger training set. This is not surprising, given that many MWEs exhibit an irregular syntactic behavior that might even seem strange at times. For example, in the sentence fragment "and then he came to.", 'came to' is an MWE. A parser might find it difficult to parse the sentence correctly, expecting a noun phrase to follow the 'to'.

Second, as described above, the syntactic features consist of general syntactic relations extracted from the parse tree and not type-specific knowledge. As a result, literal or non-expression uses of the MWE's components, which have a close syntactic relation in a given sentence, appear as non-compositional uses of the expression to the classifier.

6 Discussion

This study has addressed MWE identification: deciding if a potential use of an expression is a non-compositional one. Despite its importance in basic NLP tasks, the problem has been largely overlooked in NLP research, probably due to it presumed simplicity. However, as we have shown, simple methods for MWE identification, such as our baselines, do not perform consistently well across MWEs. This study serves to highlight this point and the need for more sophisticated methods for MWE identification.

We have shown that using a supervised learning method employing surface sentence features based on canonical form, it is possible to improve performance significantly. Unlike previous research, our method is not tailored to specific MWE types, and we did not ignore non-expression uses in our experiments.

Future research should experiment with non-verbal MWEs, since our features are not specific to verbal MWE types. Another direction is a more sophisticated corpus sampling algorithm. The current work ignored MWEs which had an unbalanced training set (usually too few positives). Methods for gathering enough positive instances of such MWEs will be useful for testing the methods proposed here, as well as for general MWE research.

References

Iñki Alegria, Olatz Ansa, Xabier Artola, Nerea Ezeiza, Koldo Gojenola and Ruben Urizar. 2004. Representation and treatment of multiword expressions in Basque. *ACL '04 Workshop on Multiword Expressions.*

The British National Corpus. 2001. *The British National Corpus, version 2 (BNC World).* Distributed by Oxford University Computing Services on behalf of the BNC Consortium. URL: http://www.natcorp.ox.ac.uk/

Elisabeth Breidt, Frederique Segond, and Giuseppen Valetto. 1996. Local grammars for the description of multi-word lexemes and their automatic recognition in texts. *COMPLEX '96.* Budapest.

Paul Cook, Afsaneh Fazly, and Suzanne Stevenson. 2007. Pulling their weight: Exploiting syntactic forms for the automatic identification of idiomatic expressions in context. *ACL '07 Workshop on A Broader Perspective on Multiword Expressions.*

Collins COBUILD. 2003. *Collins COBUILD Advanced Learner's English Dictionary.* HarperCollins Publishers, 4th edition.

Paul Deane. 2005. A nonparametric method for extraction of candidate phrasal terms. *ACL '05.*

Gael Dias. 2003. Multiword unit hybrid extraction. *ACL '03 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment.*

Afsaneh Fazly and Suzanne Stevenson. 2006. Automatically constructing a lexicon of verb phrase idiomatic combinations. *EACL '06.*

Christiane Fellbaum, Alexander Geyken, Axel Herold, Fabian Koerner, and Gerald Neumann. 2006. Corpus-based studies of German idioms and light verbs. *International Journal of Lexicography*, 19(4):349–361.

Christiane Fellbaum. 1998. Towards a representation of idioms in WordNet. *COLING-ACL '98 Workshop on the Use of WordNet in Natural Language Processing Systems.*

Nicole Grégoire. 2007. Design and implementation of a lexicon of Dutch multiword expressions. *ACL '07 Workshop on A Broader Perspective on Multiword Expressions.*

Chikara Hashimoto, Satoshi Sato, and Takehito Utsuro. 2006. Japanese idiom recognition: Drawing a line between literal and idiomatic meanings. *COLING-ACL '06, Poster Sessions.*

John S. Justeson and Slava M. Katz. 1995. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1:9–27.

Graham Katz and Eugenie Giesbrecht. 2006. Automatic identification of non-compositional multiword expressions using latent semantic analysis. *COLING-ACL '06 Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties..*

Geoffrey Leech, Roger Garside and Michael Bryant. 1994. CLAWS4: The tagging of the British National Corpus. *COLING '94.*

Marie-Catherine de Marneffe, Bill MacCartney and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. *LREC '06.*

Rosamund Moon. 1998. *Fixed Expressions and Idioms in English.* Oxford: Clarendon Press.

John Platt. 1998. Machines using sequential minimal optimization. In *In B. Schoelkopf and C. Burges and A. Smola, editors, Advances in Kernel Methods – Support Vector Learning.*

Susanne Z. Riehemann. 2001. *A Constructional Approach to Idioms and Word Formation*. Ph.D. Thesis. Stanford.

Caroline Sporleder and Linlin Li. 2009. Unsupervised recognition of literal and non-literal use of idiomatic expressions. *EACL '09.*

Aline Villavicencio, Ann Copestake, Benjamin Waldron, and Fabre Lambeau. 2004. Lexical encoding of MWE. *ACL '04 Workshop on Multiword Expressions*.

Joachim Wermter and Udo Hahn. 2004. Collocation extraction based on modifiability statistics. *COLING '04.*

Joachim Wermter and Udo Hahn. 2006. You can't beat frequency (unless you use linguistic knowledge) – a qualitative evaluation of association measures for collocation and term extraction. *COLING-ACL '06*.

Ian H. Witten amd Eibe Frank. 2000. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann.