A data-driven approach to verbal multiword expression detection. PARSEME Shared Task system description paper

Tiberiu Boroș, Sonia Pipa, Verginica Barbu Mititelu, Dan Tufiș
Research Institute for Artificial Intelligence ”Mihai Drăgănescu”,
Romanian Academy,
Bucharest, Romania
{tibi,sonia,vergi,tufis}@racai.ro

Abstract

Multiword expressions are groups of words acting as a morphologic, syntactic and semantic unit in linguistic analysis. Verbal multiword expressions represent a subgroup of multiword expressions, namely that in which a verb is the syntactic head of the group considered in its canonical (or dictionary) form. All multiword expressions are a great challenge for natural language processing, but the verbal ones are particularly interesting for tasks such as parsing, as the verb is the central element in the syntactic organization of a sentence. In this paper we introduce our data-driven approach to verbal multiword expressions, which was objectively validated during the PARSEME shared task on verbal multiword expressions identification. We tested our approach on 12 languages, and we provide detailed information about corpora composition, feature selection process, validation procedure and performance on all languages.

1 Introduction

The term “multiword expressions” (MWEs) denotes a group of words that act as a morphologic, syntactic and semantic unit in linguistic analysis: their linguistic behavior (inflection, combination with other words, meaning) cannot be inferred from the characteristics of their components. As the name suggests, verbal MWEs (VMWEs) require the presence of a verb head in the prototypical form of the MWE. The importance of identifying MWEs in natural language processing, as well as the appropriate techniques to deal with this linguistic phenomenon were discussed by (Sag et al., 2002), among others. VMWEs are particularly important for parsing, mainly because the verb is the central element in the syntactic organization of a sentence.

For the present task we focused on both detection and type-labeling of VMWEs. Though similar in nature, detection and type-labeling require different training strategies, at least in the fine-tunning stage of the system. In our case, this meant that the two tasks might require different context windows and feature sets (see Section 3 for more details). Moreover, though we applied our system on twelve languages, we performed fine-tunning of the parameter set only for the Romanian corpus (due to time constraints) and we used the same parameter set for all languages. However, the proposed fine-tunning strategy can be applied on any dataset and, in the future, we plan to make language-dependent optimization and re-run the MWE detection and labeling process for each language with its own parameters.

2 Corpora composition

During the system preparation for the PARSEME shared task on VMWEs identification (Savary et al., 2017) we were granted access to training data in the form of annotated text for 18 languages. The annotation was provided using a custom designed format called parsemetsv[1] (one-token per line with tokenization and VMWEs information, stored as tab-separated values). For some languages, lemmatization and tagging information was provided in CONLL format[2].

From the 18 languages we focused on a subset of 12 languages, because both parsemetsv information and morphosyntactic analysis were pro-

[1] http://typo.uni-konstanz.de/parseme/index.php/2-general/184-parseme-shared-task-format-of-the-final-annotation (last accessed 2017-01-29)

[2] http://universaldependencies.org/format.html (last accessed 2017-01-29)

Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017), pages 121–126,
Valencia, Spain, April 4. ©2017 Association for Computational Linguistics

vided for them: RO, FR, CS, DE, EL, ES, HU, IT, MT, SL, SV and TR. The Farsi and and Polish corpora were also provided with all the necessary information, but due to technical difficulties, we were unable to cope with the file encodings before the submission deadline and we were unable to provide an accurate evaluation on these languages.

Regarding granularity, 5 VMWE classes are used in the annotation process:

- **Ligth Verb Constructions (LVC)**: they are made up of a verb and a noun: the former has little if any semantic content, while the latter contributes the semantics of the VMWE;
- **Idioms (ID)**: these are expressions in which the verb can combine with various other words and their key-characteristic is the lack of compositional meaning;
- **Inherently reflexive verbs (IReflV)**: they are made up of a verb and a reflexive clitic and their meaning is different from those occurrences of the verb without the clitic (in case this is possible); the passive, reciprocal, possessive and impersonal constructions are excluded from annotation;
- **Verb-Particle Constructions (VPC)**: they contain a verb and a particle and have a non-compositional meaning;
- **Other (OTH)**: any VMWE that does not fit any of the above mentioned classes.

The LVC and ID categories are considered universal, in the sense that they apply to all languages involved in the shared task[3], whereas IReflV applies to all Romance languages, to all Germanic languages in the shared task and almost all Balto-Slavic ones (the exception is Lithuanian). VPC applies to all Germanic languages, to Italian, Slovene, Greek, Hebrew and Hungarian. Except for Lithuanian, OTH can occur in any language in the task, although not necessarily present in the data.

The distribution of these categories over the training sets for the languages considered here is given in Table 1 below.

3 Sequence labeling for verbal multiword expression detection

When it comes to automatic identification of VMWEs, aside from rule-based approaches such as tree substitution grammars (Green et al., 2011) and dependency lexicons (Bejcek et al., 2013), several research have addressed statistical methods. These statistical methods refer to n-gram based approaches (Pedersen et al., 2011), Latent Semantic Analysis (LSA) (Katz and Giesbrecht, 2006), word association measures (Pecina, 2008) and many classification-based approaches.

In our approach, which is also a statistical method, we treat VMWEs identification as a sequence labeling approach, in which we employ a Conditional Random Fields (CRF) classifier (Lafferty et al., 2001) trained to predict transitions between labels rather than the labels themselves. For every word inside a sentence we trained the classifier to predict a label using lemma and part-of-speech based features for a window of words centered on the current position. A naive method would use the VMWE type as labels and employ a dummy label for words that do not belong to any unit. However, a more principled approach is to perform VMWE identification in two steps:

- **Head labeling**: in this step we identify words that introduce VMWEs, a good choice for these words being the verb, in head-initial languages.
- **Tail labeling**: in this step we identify the words that link to the head word and contribute to the unit.

Our experiments showed that when the head of a MWE is correctly identified, the linking of the other constituents of the MWE is easier. This reflected also in the fine tuning of the two distinct phases: the head of a MWE was identified using two-word windows and the L+P set of parameters (see section 3) while the linking phase relied on 4-word windows with the same parameters. This two-step approach increased of precision by 9%. Thus we considered that that the two-step approach works significantly better than the one-shot detection and labeling of VMWEs. As mentioned, the two-step approach uses different feature windows for head and tail identification. The larger window (used in tail identification) pro-

[3] Although considered applicable, the LVC category did not occur in the Farsi data, while ID did not occur in the Farsi or Hungarian data.

VMWE type	CS	DE	EL	ES	FR	HU	IT	MT	RO	SL	SV	TR
IReflV	8851	111	0	336	1313	0	580	0	2496	945	3	0
LVC	2580	178	955	214	1362	584	395	434	1019	186	13	2624
ID	1419	1005	515	196	1786	0	913	261	524	283	9	2911
VPC	0	1143	32	0	0	2415	62	0	0	371	31	0
OTH	2	10	16	2	1	0	4	77	15	2	0	634

Table 1: VMWE distribution in the training corpora for the 12 languages

ved to be inefficient for head labeling, but provided better results in the second step [4].

The training data contained several overlapped VMWEs. In theory, our proposed labeling scheme should be able to handle such cases (i.e., if a head token is also linked as a tail, then that token and its tail should be embedded in the higher VMWE). However, because of their sparseness in the training data, our system did not spot such cases.

4 Validation and feature selection procedure

All our results are reported for a 10-fold validation procedure, which takes into account the distribution of VMWE types in the training corpora. This means that when we split our data into 90% training and 10% validation we strived to preserve the relative distribution of labels in order to report results as close as possible to real-life data.

4.1 Head labeling

After a shallow investigation of different feature sets we established that lemma, part-of-speech (POS) (with attributes) and a combined feature from lemma+POS are the best candidates for fine-tuning. This first feature set is denoted as L+P. We tried to extend this setup by adding 4 new features (whenever possible): gender, person, number and a special flag for reflexive pronouns (L+P+E). In Table 2 we show the detailed results obtained on the Romanian training corpus using the two feature sets (L+P and L+P+E) and varying the feature window size, in the 10-fold validation procedure.

As can easily be seen, the overall F-score of the system decreases for feature windows higher than 2, which indicates over-fitting of the training data. Also, for the window size of 2 the extended feature set provides a better precision but decreases the recall, yielding in a lower F-score. Thus, our final choice was a window size of two with the L+P feature set.

W	Feat-set	P	R	F
2	L+P	0.8957	0.8952	**0.8914**
	L+P+E	**0.9012**	0.8769	0.8889
3	L+P	0.8912	0.8842	0.8877
	L+P+E	0.8778	0.8868	0.8823
4	L+P	0.8656	0.8869	0.8761
	L+P+E	0.8378	0.8845	0.8605

Table 2: Results on the training set

4.2 Tail labeling

Tail labeling is carried out on an extended feature set in which we added additional information about labels previously assigned during head labeling. Our experiments showed that varying the feature window has little impact on the system's performance and we decided to use a feature window of 4 (totally, 9 words).

In Table 6, for head labeling, the first column represents the words lemmas, the second column contains the part-of-speech with its associated attributes and the third column is used for the label itself. Note that during head labeling we ignore any linked words. Next, for tail labeling we extend the feature-set and we add one column, which is used for head labels. In the training phase we use the head-labels extracted from the training corpus and at runtime we use the classifier to predict these labels in the first phase of the two-step approach.

In the template file[5] (Table 8), each line starts with a string that uniquely identifies the feature (i.e., "U01", "U02", etc.). Next to the identifier we can add any feature (%x) and any combination of features ('/' is used for combining multiple features). Features in the training data are ex-

[4] In the feature selection process, described in the next section, we found that the best results are obtained using a feature-window of two (totally, 5 words included) for head labeling and a window of 4 (totally, 9 words included) for tail labeling

[5] standard CRF++ (https://taku910.github.io/crfpp/) template file

Label	P	Stdev	R	Stdev	F-score	Stdev
ID	0.8760	0.0434	0.6421	0.0727	0.7398	0.0612
IReflV	0.8830	0.0207	0.9611	0.0129	0.9202	0.0113
LVC	0.9363	0.0219	0.8590	0.0322	0.8955	0.0202
PREV	0.9837	0.0087	0.9655	0.0105	0.9745	0.0068

Table 3: Detailed results for Romanian reported for every VMWE type using 10- fold validation. The 'PREV' label is used for tail linking

CM	_	IReflV	ID	LVC
_	-	8	13	15
IReflV	38	239	0	0
ID	2	0	37	3
LVC	3	1	0	84

Table 4: Confusion matrix computed for the first fold of the RO corpus. Symbol '_' is used to denote dummy tokens - token does not belong to any VMWE

	Strict			
Lang	**P**	**R**	**F**	**Rank**
CS	0.7009	0.5918	0.6418	2/4
DE	0.3652	0.13	0.1917	4/4
EL	0.4286	0.252	0.3174	2/4
ES	0.6447	0.196	0.3006	4/4
FR	0.7415	0.35	0.4755	3/5
HU	0.8029	0.5471	0.6508	3/4
IT	0.6125	0.098	0.169	3/3
MT	0.2333	0.028	0.05	3/3
RO	0.8652	0.706	0.7775	1/4
SL	0.5503	0.208	0.3019	4/4
SV	0.5758	0.161	0.2517	3/3
TR	0.6304	0.4391	0.5176	2/4
	Fuzzy			
Lang	**P**	**R**	**F**	
CS	0.819	0.6228	0.7076	3/4
DE	0.6716	0.1793	0.283	4/4
EL	0.5616	0.2953	0.3871	4/4
ES	0.7233	0.1967	0.3093	4/4
FR	0.7872	0.3673	0.5009	3/4
HU	0.8208	0.5015	0.6226	4/4
IT	0.6837	0.1053	0.1824	3/3
MT	0.2481	0.0259	0.0469	3/3
RO	0.8773	0.7019	0.7799	4/4
SL	0.7339	0.2145	0.332	4/4
SV	0.6538	0.1677	0.2669	3/3
TR	0.634	0.4348	0.5159	3/4

Table 5: Evaluation campaign results

Head labeling			
Portugalia	Np	_	
s	Ncmprn	IReflV	
-	DASH	_	
avea	Vaip3s	_	
confrunta	Vmp	_	
cu	Sp	_	
același	Dd3fsr	_	
situație	Ncfsrn	_	
:	COLON	_	
Tail labeling			
Portugalia	Np	_	_
s	Ncmprn	IReflv	_
-	DASH	_	_
avea	Vaip3s	_	_
confrunta	Vmp	_	PREV
cu	Sp	_	_
același	Dd3fsr	_	_
situație	Ncfsrn	_	_
:	COLON	_	_

Table 6: Excerpt from the training data - Romanian version of the training corpus

tracted using a "relative coordinate systems". The first coordinate is the relative row index, and the second one is the 0-indexed absolute column position of the feature. For instance, x[-1,1] signifies the lemma (1 - second column) of the previous token (-1 - the above row).

Head labeling template file

```
U01:%x[0,0]
U02:%x[0,1]
U03:%x[0,0]/%x[0,1]

U04:%x[-1,0]
U05:%x[-1,1]
U06:%x[-1,0]/%x[0,1]
...
3 more similar feature sets
```

Tail labeling template file

```
U01:%x[0,0]
U02:%x[0,1]
U03:%x[0,2]
U04:%x[0,0]/%x[0,1]
...
8 more similar feature sets
```

Table 8: The template file used with the CRF++ classifier

Language	**Type**	**P**	**R**	**F**	**Language**	**P**	**R**	**F**
CS	**LVC**	**0.7460**	**0.2741**	**0.4009**	**DE**	0.0000	0.0000	0.0000
	IReflV	**0.7109**	**0.7554**	**0.7325**		**0.4000**	**0.1000**	**0.1600**
	VPC	N/A	N/A	N/A		**0.6667**	0.1593	0.2571
	ID	**0.5909**	**0.1354**	**0.2203**		**0.3433**	**0.1075**	**0.1637**
EL	**LVC**	**0.4096**	**0.2798**	**0.3316**	**ES**	**0.6111**	0.2018	**0.3034**
	IReflV	N/A	N/A	N/A		**0.6559**	0.2735	0.3861
	VPC	**0.6667**	**0.2500**	**0.3636**		N/A	N/A	N/A
	ID	**0.2321**	0.1024	0.1421		0.0000	0.0000	0.0000
FR	**LVC**	**0.7255**	0.1365	0.2298	**HU**	**0.6383**	0.2055	0.3109
	IReflV	**0.7000**	0.6667	**0.6829**		N/A	N/A	N/A
	VPC	N/A	N/A	N/A		**0.8294**	0.6864	0.7512
	ID	**0.7294**	**0.5210**	**0.6078**		N/A	N/A	N/A
IT	**LVC**	**0.7000**	0.0805	**0.1443**	**MT**	**0.1837**	**0.0347**	**0.0584**
	IReflV	0.3636	0.0533	0.0930		N/A	N/A	N/A
	VPC	0.3333	0.0909	0.1429		N/A	N/A	N/A
	ID	**0.6667**	**0.1200**	**0.2034**		**0.2000**	**0.0108**	**0.0205**
RO	**LVC**	**0.9167**	**0.8148**	**0.8627**	**SL**	**0.6667**	**0.0444**	**0.0833**
	IReflV	**0.8197**	0.6897	0.7491		**0.5390**	0.3004	0.3858
	VPC	N/A	N/A	N/A		**0.6757**	**0.2315**	**0.3448**
	ID	**0.8864**	**0.5200**	**0.6555**		**0.5000**	0.0109	0.0213
SV	**LVC**	**0.4000**	0.1429	**0.2105**	**TR**	**0.6797**	**0.5226**	**0.5909**
	IReflV	0.0000	0.0000	0.0000		N/A	N/A	N/A
	VPC	**0.5614**	**0.2065**	**0.3019**		N/A	N/A	N/A
	ID	**0.5000**	**0.0196**	**0.0377**		**0.5921**	**0.3614**	**0.4489**

Table 7: Strict evaluation results for VMWE type identification. Best results in the challenge are BOLD

4.3 Further discussion of the results

The values reported in Table 2 refer to the overall performance of the system, regardless of the VMWE class. In order to offer a better view on the system performance we provide accuracy figures for every VMWE class (Table 3), as well as the confusion matrix for head labeling computed on the first training fold of the validation (Table 4).

As shown in the confusion matrix, the system rarely confuses one VMWE for another, most errors being omissions - head VMWE tokens being labeled with "_" (dummy) labels. While IReflVs are both numerous and easy to spot, IDs are rare and extremely difficult to label because their identification involves semantics as well as syntactic knowledge. The IDs correctly spotted by the system in this fold may have been "over-fitted" during the training. However, it is highly possible that, with another corpus, ID identification fail mainly because of the ambiguities that arise when trying to determine if the "sum" of the words senses is different from the VMWE sense (a task which is barely handled by the CRF and feature set combination).

5 Results and conclusions

The final evaluation results that we report in this paper are the results obtained during the PARSEME shared task on VMWE identification. As previously mentioned, we trained and submitted runs for 12 languages (table 5 summarizes the results)[6]. We must mention that for the shared task, VMWE type identification was not mandatory. However, we as well as three other teams included this information in their submissions. As such, we show detailed results for each VMWE class in Table 7, where we give the results for the strict evaluation.

For Romanian, there is a notable difference in the F-score reported during 10-fold validation and PARSEME evaluation, which is caused mainly by the skewed distribution of VMWE types in the test data. However, the F-score reported for individual VMWE classes are well within the standard deviation computed in table 3. Similar conditions may apply to the other languages. Also, as previously stated, our fine-tunning process was only performed on the Romanian dataset, where we obtained the highest score in the strict evaluation of the system. An identical process can be carried out on any dataset and for best results, one would have to perform this tunning in order to obtain language-dependent optimizations.

The system is freely available and can be obtained by contacting the authors.

[6] http://typo.uni-konstanz.de/parseme/index.php/2-general/142-parseme-shared-task-on-automatic-detection-of-verbal-mwes - accessed 2017-02-15

References

Eduard Bejcek, Pavel Stranák, and Pavel Pecina. 2013. Syntactic identification of occurrences of multiword expressions in text using a lexicon with dependency structures. In *Proc. of the 9th Workshop on Multiword Expressions*, pages 106–115.

Spence Green, Marie-Catherine De Marneffe, John Bauer, and Christopher D Manning. 2011. Multiword expression identification with tree substitution grammars: A parsing tour de force with french. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 725–735. Association for Computational Linguistics.

Graham Katz and Eugenie Giesbrecht. 2006. Automatic identification of non-compositional multi-word expressions using latent semantic analysis. In *Proceedings of the Workshop on Multiword Expressions: Identifying and Exploiting Underlying Properties*, pages 12–19. Association for Computational Linguistics.

John Lafferty, Andrew McCallum, Fernando Pereira, et al. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1, pages 282–289.

Pavel Pecina. 2008. A machine learning approach to multiword expression extraction. In *Proceedings of the LREC Workshop Towards a Shared Task for Multiword Expressions (MWE 2008)*, volume 2008, pages 54–61. Citeseer.

Ted Pedersen, Satanjeev Banerjee, Bridget T McInnes, Saiyam Kohli, Mahesh Joshi, and Ying Liu. 2011. The ngram statistics package (text:: nsp): A flexible tool for identifying ngrams, collocations, and word associations. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real World*, pages 131–133. Association for Computational Linguistics.

Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger, 2002. *Multiword Expressions: A Pain in the Neck for NLP*, pages 1–15. Springer Berlin Heidelberg, Berlin, Heidelberg.

Agata Savary, Carlos Ramisch, Silvio Cordeiro, Federico Sangati, Veronika Vincze, Behrang QasemiZadeh, Marie Candito, Fabienne Cap, Voula Giouli, Ivelina Stoyanova, and Antoine Doucet. 2017. The PARSEME Shared Task on Automatic Identification of Verbal Multiword Expressions. In *Proceedings of the 13th Workshop on Multiword Expressions*, Valencia, Spain, April. Association for Computational Linguistics.