Processing European Portuguese Verbal Idioms: From the Lexicon-Grammar to a Rule-based Parser*

Ana Galvão[1,3 [0000−0002−0045−012X]], Jorge Baptista[2,3 [0000−0003−4603−4364]], and Nuno Mamede[1,3 [0000−0001−6033−158X]]

[1] Instituto Superior Técnico, Universidade de Lisboa
Av. Rovisco Pais 1, P-1049-001 Lisboa, Portugal,
a.s.galvao@tecnico.ulisboa.pt
[2] Universidade do Algarve - FCHS, Campus de Gambelas, P-2005-139 Faro, Portugal,
jbaptis@ualg.pt
[3] L2F - Spoken Language Laboratory, R. Alves Redol 8, P-1000-029 Lisboa, Portugal,
Nuno.Mamede@tecnico.ulisboa.pt

Abstract. Processing verbal idioms is a challenging task for Natural Language Processing systems because they are syntactically analysable strings, with a well-formed structure, identical to that of distributionally free sentences, but whose meaning is for the most part non-compositional. This paper presents recent advances in processing European Portuguese verbal idioms. From a lexicon-grammar matrix, containing +2,500 verbal idioms and +100 (structural, distributional and transformational) properties, parsing rules are automatically generated, within the framework of a rule-based incremental parser. They are then integrated in STRING, a fully-fledged natural language processing system for Portuguese. The system now identifies not only the idioms' base forms, but also the sentences resulting from some productive and very general transformations (passive, pronominalisation), admitted by some of these idioms. Other improvements include: a newly developed lexicon-grammar *validator*, a new *generation module* for transformations' examples, and a new, more granular, *evaluation* module. An *intrinsic* evaluation achieves an overall recall of 92.5%.

Keywords: Verbal idioms · Frozen sentences · European Portuguese · Lexicon-Grammar · Natural Language Processing.

1 Introduction

Verbal idioms (e.g. *não mexer um palha* lit.:'do not move a straw', 'be idle or indifferent') are a type of *frozen sentences* where the verb and at least one of its arguments are frozen together. By 'frozen' we mean that strong combinatorial constraints can be observed between the verb and at least one of its arguments. The syntactic properties and the overall meaning of the idiom cannot be derived from properties and the individual meaning of its component elements, when they are used independently. Therefore,

* Research for this paper was partially supported by national funds through Fundação para a Ciência e a Tecnologia (ref. UID/CEC/50021/2019).

EUROPHRAS 2019, pages 70–77,
Malaga, Spain, September 25-27, 2019.
https://doi.org/10.26615/978-2-9701095-6-3_009

this information must be encoded in the lexicon, to be precise, in the *lexicon-grammar* of the language [8,9]. In this theoretical and methodological framework, the meaning unit is not the word but the *elementary sentence*, in this case, the verbal idiom, along with its relevant syntactic-semantic (i.e. distributional, structural and transformational) properties.

Verbal idioms can be construed as a special type of *multiword expressions* [5] and constitute a large set of the lexicon-grammar of many languages [10,11], though their frequency in texts is often very low. Processing verbal idioms is a challenging task for Natural Language Processing (NLP) systems [18] because they are syntactically analysable strings, with a well-formed structure, identical to that of distributionally free sentences, but whose meaning is for the most part non-compositional. Processing multiword expressions, including verbal idioms, is essencial to represent the meaning of a text in a adequate way. The low frequency of many verbal idioms in corpora makes spotting them a difficult task [13], and much prior has been dedicated to identifying them in texts [14,15,16]. However, the focus of this paper will not be on *identification* (in a lexicographic perspective), but rather on the *processing* of an *already built* computational lexicon (a lexicon-grammar) of verbal idioms [2,3], particularly of transformationally-derived, equivalent sentence-forms (for lack of space, this lexicon-grammar will not be presented here). In fact, little relevance has been given to *transformations* of verbal idioms, that is, sentences that are derived from their base form by general formal changes such as *passive* or *pronominalisation* of free arguments.

This is the major contribution of this paper. It presents the improvements introduced in processing European Portuguese verbal idioms, within the development of a Portuguese NLP system, STRING [12][4], allowing it now to identify the most common transformations of verbal idioms. These improvements include: (i) a newly developed *validator* of the lexicon-grammar matrix, to help linguists represent in a consistent a systematic way the verbal idioms in this computational lexicon; (ii) a new *example generation* module, that produces natural language examples for the transformations allowed by each verbal idiom; (iii) a new *rule generation* module, automatically producing the parsing rules to identify both base sentences and their transformations in texts; and (iv) a new, more granular and expedite, *evaluation* module, for an intrinsic assessment of the system's performance.

2 Validator

Since the encoding of linguistic information is a manual procedure, which is a very time-consuming and error-prone task, a *Validator* was build, written in Perl, to check the formal consistency of the matrix. The validator takes as input the CSV-converted lexicon-grammar matrix and performs the following checks, outputting the corresponding error messages: (i) *cell content validation*: checks if the content of the cell in a given column is consistent with the predefined values for that column; (ii) *class consistency cross-validation*: depending on the classe of the idiom, the number of relevant columns can vary; the system checks the consistency of the properties with the overall class definition; (iii) related *properties cross-validation*: consistency among related properties,

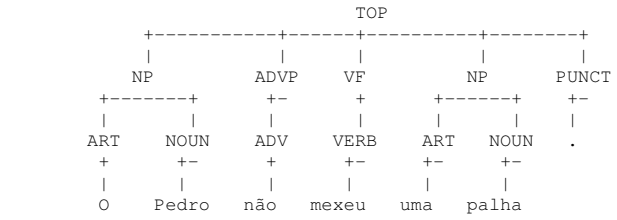
[4] `https://string.l2f.inesc-id.pt/`

represented in different columns is cross-checked. Several dozens of these rules were manually crafted. Based on the error messages outputted by the validator, it is possible to detect most input errors, which are then manually corrected.

3 Rules Generation Process

In this Section, a brief, overall view of the STRING system [12] is presented first, in order to better frame the task of syntactic analysis (parsing) and the identification of verbal idioms. The STRING system performs all basic operations in a pipeline of modules, from tokenization, text segmentation (sentence splitting) and part-of-speech (PoS) tagging [7,20], and rule-based and statistical PoS disambiguation [6,7]. The syntactic analysis is carried out by *XIP*, the Xerox Incremental Parser [1], using a rule-based grammar specifically built to process Portuguese. It is in this module that verbal idioms are processed. The parser processes texts sentence by sentence. Firstly, it groups words into elementary syntactic phrases or *chunks*, such as noun phrases (`NP`); this *shallow parsing* operation is called *chunking*. Then, the system extracts syntactic dependencies between the chunks' heads, e.g. the `SUBJ` (subject); a `CDIR` (direct complement) dependency between a NP's head and a verb; and an "umbrella" dependency `MOD` (modifier) linking the nominal head of a `PP` to a verb (or an adjective to a noun, or an adverb to verb).

This is the output of a sentence with the verbal idiom *não mexer um palha* 'be idle or indifferent'. Each word is associated to a part-of-speech node and then grouped into chunks: `NP` (noun phrases, twice), `ADVP` (adverbial phrase) and `VF` (finite verb phrase). Bellow the chunk tree, the list of extracted dependencies is presented.




```
MAIN(mexeu)             VDOMAIN(mexeu,mexeu)     MOD_PRE_NEG(mexeu,não)
QUANTD_PRE(palha,uma)   SUBJ_PRE(mexeu,Pedro)    NE_PEOPLE_INDIVIDUAL(Pedro)
DETD(Pedro,O)           CDIR_POST(mexeu,palha)   FIXED(não,mexeu,palha)
0>TOP{NP{O Pedro} ADVP{não} VF{mexeu} NP{uma palha} .}
```

Since the syntactic structure of verbal idioms is by and large identical to that of ordinary, distributionally free verbs; and since even very general transformations, such as the passive or the pronominalization, can be allowed in some cases; we adopted the strategy, as presented in [3,4,17], to first allow the parser to perform a general-purpose analysis of the sentences, as shown above; and, then, use the result of this parse to capture the verbal idiom lexical-syntactic pattern. This is represented by another dependency, `FIXED`.The (fully expanded) rule for this idiom is the following:

```
if ( VDOMAIN(#?,#2[lemma:mexer]) & MOD[neg,pre](#2,#3) & CDIR[post](#2,#4[surface:palha])
     & QUANTD(#4,?[surface:uma]) )
    FIXED(#3, #2, #4)
```

The rule first matches a main verb (VDOMAIN) *mexer* 'move' with a negation (neg) modifier and a direct complement (CDIR) *palha* 'straw' and then produces the FIXED dependency. A rule is automatically generated for each verbal idiom in the matrix based solely in the information encoded therein. To do so, each relevant column value contributes with a condition to the rule (relevant columns depend on the verbal idiom class); and each main constituent's head is associated to a variable; conditions are concatenated by operator '&'.

4 Processing Transformations

This is one of the important developments introduced in this paper. In case any transformations apply, the if() structure is branched in alternative sets of conditions (noted '||'). For example, for the sentence: *O jornalista bombardeou* ***a atriz*** *com perguntas inconvenientes* 'The reporter bombarded the actress with impertinent questions', the reduction of the free direct complement to an accusative pronoun (PronA): *O jornalista bombardeou****-a*** *com perguntas inconvenientes* 'The reporter bombarded her with impertinent questions', is expressed by the rule:

```
if (  VDOMAIN(#?,#2[lemma:bombardear]) &
    ( CDIR[post](#2,#3[UMB-Human]) || CLITIC(#2,?[acc]) ) &
      MOD[post](#2,#4[surface:perguntas]) & PREPD(#4,?[surface:com]) )
   FIXED(#2, #4)
```

More precisely, the CDIR can alternate ('||') with a CLITIC dependency between the verb and a personal pronoun in the accusative case [acc] (irrespective of the pronoun being before or after the verb). A similar procedure was used for the dative (PronD) and the reflex (PronR) pronominalisations. In the case of the passive transformation, *A atriz foi bombardeada com perguntas inconvenientes pelo jornalista* 'The actress was bombarded with inconvenient questions by the reporter', the corresponding new rule is:

```
if (  VDOMAIN(#?,#2[pass-ser,lemma:bombardear]) & SUBJ(#2,?[UMB-Human]) &
      MOD[post](#2,#3[surface:perguntas]) & PREPD(#3,?[surface:com]) )
   FIXED(#2, #3)
```

The VDOMAIN condition relies on the previous identification of the passive construction pattern with auxiliary verb *ser* 'be'. A conversion table is used to associate the passive sentence constituents to the correspondent variables in the new rule.

A *configuration file* makes it possible to determine which restrictions are to be applied to the rule generation process (unless otherwise determined, fully expanded rules are produced, as shown above). The controllable restrictions apply to determiners, prepositions and both left and right modifiers of the frozen head noun; and to the distributional constraints to any of the free constituents, both the subject and/or the complements.

5 Generation of Transformation-derived Examples

The example generation process starts by verifying for each idiom whether its description in the lexicon-grammar matrix allows for any transformation and if so, the system

reads the content of each constituent. A conversion table is then used to associate the idiom's constituents to the transformed sentence constituents, which will correspond to a given set of variables in the (new) rule. For example, the direct complement (variable `#3`) of a verb (`#2`) becomes the subject (`#1`) of a passive sentence (see the passive rule above). About 1,170 transformationally-derived sentences were generated for a set of 7 transformations (4 types of pronominalization, the dative restructuring and 2 types of passive constructions), and they were manually revised by a linguist. This was done in several iterations, until satisfactory results were achieved.

6 Evaluation

A newly-built *Evaluation module* was used for an *intrinsic* evaluation of the system's performance. The evaluation uses the lexicon-grammar manually produced examples and the automatically generated, transformation-derived examples, along with the expected output for each sentence, i.e. the `FIXED` dependency with all its arguments. These examples constitute our evaluation corpus. It must be stressed that the examples were produced not taking into account any of the processing steps prior to parsing so that many types of errors may accumulate along this processing pipeline.

The new evaluation module improved the granularity of the system's evaluation concerning verbal idioms, as it considers now 3 criteria for the successful extraction of the idioms' dependency: (i) the extraction of the `FIXED` dependency; (ii) the production of the correct number of arguments for the dependency; (iii) the correct identification of the lexical elements for each argument of the dependency. In case no `FIXED` dependency is extracted, the system returns '`FAILED`'. Notice that the previous evaluation module [3] only returned whether the `FIXED` dependency was extracted or not. Besides, the new module processes all sentences in a single batch, so it takes much less time (6 min.) to obtain the results, contributing to a more efficient development of the lexicon-grammar.

Table 1 shows the results for the manually produced sentences per verbal idioms' class. Overall, recall varies from 86.8% for the more relaxed criterion of just capturing the `FIXED` dependency; to 85.3, when the number or the dependency's arguments (`NB-ARG`) is considered; down to 78.2% for a complete match of the dependency's arguments (`ARG`). It is noteworthy to mention that, though the criteria are progressively more strict, the 8.6% drop in the system's performance is relatively small. Next, Table 2 shows the results for the automatically generated, transformation-derived examples, per transformation.

Most errors were found to be due to previous steps of the processing pipeline. For example, the incorrect disambiguation of *a*, either as the definite article 'the' (fem. sg.) or as the preposition *a* 'to'; or the incorrect attribution of lemma to ambiguous verb forms (*foi*: *ser/ir* 'be/go'). Compound (or multiword) lexical units also hinder the process, as the system gives precedence to them (e.g. *por conta de* 'because' vs. *A Ana vive* ***por conta de*** *_o Pedro* 'Ana depends on Pedro for a living'). Few errors were found due to parsing. For example, the chunking rules failed to produce a `PP` for the sequence *entre nós* 'among us' in the idiom *O Pedro já não está* ***entre nós***, lit. 'Pedro is no longer among us', 'Pedro has died'. Many of these errors have been corrected since, either by

Table 1. Results for verbal idioms' identification: manually produced sentences.

Class	Total	#FIXED	%	#NB-ARG	%	#ARG	%
CADV	16	7	43.8	7	43.8	5	31.3
C0	21	15	71.4	15	71.4	12	57.1
C1	503	484	96.2	481	95.6	448	89.1
CAN	182	156	85.7	156	85.7	153	84.1
CDN	46	37	80.4	37	80.4	36	78.3
C1P2	291	274	94.2	266	91.4	228	78.4
C1PN	259	224	86.5	216	83.4	206	79.5
CNP2	176	152	86.4	151	85.8	149	84.7
CP1	718	635	88.4	628	87.5	558	77.7
CPN	106	74	69.8	71	67.0	63	59.4
CPP	195	130	66.7	126	64.6	115	59.0
CPPN	36	28	77.8	27	75.0	26	72.2
CV	12	6	50.0	4	33.3	4	33.3
TOTAL	2,561	2,222	86.8	2,185	85.3	2,003	78.2

Table 2. Results for verbal idioms' identification: Automatically generated, transformation-derived sentences.

Transformation	Total	#FIXED	%	#NB-ARG	%	#ARG	%
PronA	187	170	90.9	169	90.4	165	88.2
PronD	178	131	73.6	130	73.0	129	72.5
PronPos	324	268	82.7	266	82.1	265	81.8
Rdat	192	107	55.7	106	55.2	106	55.2
PassSer	185	142	76.8	141	76.2	139	75.1
PassEstar	83	70	84.3	69	83.1	68	81.9
Total	1,170	909	77.7	902	77.1	884	75.6

improving the system's general parsing rules, e.g. the PP *entre nós* 'among us' chunking rule; or by manually producing alternative rules to the rule generation module in order to take into account multiword lexical units, e.g. *viver por conta de* 'depend on' (this often entailed changing the verbal idiom's class); or simply by using a non-ambiguous verb form in the examples. Naturally, after this process, the overall results, after a 2^{nd} run evaluation, improved significantly, as shown in Table 3 (the system current status).

7 Conclusion and Future Work

This paper expanded previous work on the automatic processing of European Portuguese verbal idioms. The lexicon-grammar of verbal idioms, a matrix with the linguistic description +2,500 frozen sentences, represented by +100 distributional, structural and transformational properties, is automatically converted into the corresponding parsing rules, so that the system may identify these idiomatic expressions in texts.

Table 3. Results for the 2^{nd} run evaluation.

Sentences	Count	FIXED	%	NB-ARG	%	ARG	%
base	2,542	2,429	0.956	2,400	0.944	2,337	0.919
transformed	1,157	1,088	0.940	1,083	0.936	1,083	0.936
Total	3,699	3,517	0.951	3,483	0.942	3,420	0.925

Each idiom is provided with a manually produced example, to illustrate that construction's base form. The paper introduced several new developments, foremost producing rules that capture the expressions derived from the base forms of those idioms by application of very general transformations (pronominalization, dative restructuring and passive constructions). The system also generates, for these transformationally derived sentences, natural language examples (1,157), which can then be used to test the system's performance. An *intrinsic* evaluation was carried out, and has shown very positive results: for the strictest criterion, an overall recall of 78.2% and 75.6% for the manually produced and for the automatically generated sentences, respectively. Most errors are due to the previous modules of the processing pipeline, particularly the PoS tagger and the statistical and rule-based disambiguator. Manual correction of these errors made it possible to achieve, in the strictest criterion, a recall of 91.9% and 93.6% for each type of examples, and an overall perfomance of 92.5% In the near future, we intend to integrate the corresponding lexicon-grammar of Brazilian Portuguese [19] and perform an *extrinsic* evaluation using (or adapting) the Portuguese corpus developed in-house [3] and that built for the PARSEME project [5] [15,16].

References

1. Ait-Mokhtar, S., Chanod, J., Roux, C.: Robustness beyond shallowness: incremental dependency parsing. Natural Language Engineering **8**(2/3), 121–144 (2002)
2. Baptista, J., Correia, A., Fernandes, G.: Frozen Sentences of Portuguese: Formal Descriptions for NLP. In: Workshop on Multiword Expressions: Integrating Processing (in EACL 2004). pp. 72–79 (2004)
3. Baptista, J., Fernandes, G., Talhadas, R., Dias, F., Mamede, N.: Implementing European Portuguese Verbal Idioms in a Natural Language Processing System. In: Corpas Pastor, G. (ed.) Computerised and Corpus-based Approaches to Phraseology: Monolingual and Multilingual Perspectives (Proceedings of EUROPHRAS 2015). pp. 102–115 (2016)
4. Baptista, J., Mamede, N., Markov, I.: Integrating verbal idioms into an NLP system. In: Computational Processing of the Portuguese Language (PROPOR 2014). LNAI/LNCS, vol. 8775, pp. 251–256. Springer (2014)
5. Constant, M., Eryigit, G., Monti, J., van der Plas, L., Ramisch, C., Rosner, M., Todirascu, A.: Multiword expression processing: A survey. Computational Linguistics **43**(4), 837–892 (2017)
6. Diniz, C.: RuDriCo2 - Um Conversor Baseado em Regras de Transformação Declarativas. Master's thesis, Universidade Técnica Lisboa - IST (2010)

[5] `https://typo.uni-konstanz.de/parseme`

7. Diniz, C., Mamede, N., Pereira, J.D.: RuDriCo2 - a faster disambiguator and segmentation modifier. In: Simpósio de Informática - INForum. pp. 573–584. Universidade do Minho, Portugal (2010)
8. Gross, M.: Une classification des phrases ≪figées≫ du français. Revue Québécoise de Linguistique **11-2**, 151–185 (1982)
9. Gross, M.: Lexicon-grammar. In: Brown, K., Miller, J. (eds.) Concise Encyclopedia of Syntactic Theories, pp. 244–259. Pergamon, Cambridge (1996)
10. Lamiroy, B.: Le lexique-grammaire: essai de synthèse. Travaux de Linguistique **37**, 7–23 (1998)
11. Lamiroy, B. (ed.): Les expressions verbales figées de la francophonie: Belgique, France, Québec et Suisse. OPHRYS, Paris (2010)
12. Mamede, N., Baptista, J., Diniz, C., Cabarrão, V.: STRING - A Hybrid Statistical and Rule-based Natural Language Processing Chain for Portuguese. In: Abad, A. (ed.) International Conference on Computational Processing of Portuguese (PROPOR 2012) - Demo Session (2012), http://www.inesc-id.pt/ficheiros/publicacoes/8578.pdf
13. Manning, Chris; Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA, 1st edn. (May 1999)
14. Pecina, P.: Lexical association measures and collocation extraction. Language Resources and Evaluation **44**, 137–158 (2010)
15. Ramisch, C., et al.: Edition 1.1 of the PARSEME Shared Task on Automatic Identification of Verbal Multiword Expressionshared Task on Automatic Identification of Verbal Multiword Expressions. In: Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018). pp. 222–240. ACL (2018), https://www.aclweb.org/anthology/W18-4925
16. Ramisch, C., Ramisch, R., Zilio, L., Villavicencio, A., Cordeiro, S.: A Corpus Study of Verbal Multiword Expressions in Brazilian Portuguese. In: Computational Processing of the Portuguese Language (PROPOR 2018). LNAI/LNCS, vol. 11122, pp. 24—34 (2018)
17. Rassi, A., Santos-Turati, C., Baptista, J., Mamede, N., Vale, O.: The fuzzy boundaries of operator verb and support verb constructions with *dar* "give" and *ter* "have" in Brazilian Portuguese. In: Proceedings of the Workshop on Lexical and Grammatical Resources for Language Processing (LG-LP2014 in COLING 2014). pp. 92–101. ACL (2014)
18. Sag, I.A., Baldwin, T., Bond, F., Copestake, A., Flickinger, D.: Multiword Expressions: A Pain in the Neck for NLP. In: Proceedings of Computational Linguistics and Intelligent Text Processing. LNAI/LNCS, vol. 2276, pp. 1–15. 3rd International Conference CICLing-2002, Springer, Berlin (2002)
19. Vale, O.A.: Expressões Cristalizadas do Português do Brasil: uma proposta de tipologia. Tese de Doutorado, Universidade Estadual Paulista, Araraquara (2001)
20. Vicente, A.: LexMan: um Segmentador e Analisador Morfológico com Transdutores. Master's thesis, Instituto Superior Técnico, Universidade de Lisboa, L^2F/INESC-ID, Lisboa, Portugal (2013)