Lexical noun phrase chunking with Universal Dependencies for Portuguese

Aleksander Tomaz de Souza[2]
Evandro Eduardo Seron Ruiz[1,2]

[1] Center for Artificial Intelligence (C4AI) – INOVA.USP
Butantã, São Paulo, SP – Brasil

[2]Departamento de Computação e Matemática
Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto (FFCLRP)
Universidade de São Paulo – USP
Ribeirão Preto, SP – Brasil

[aleksander, evandro]@usp.br

Abstract. *Partial parsing retrieves a limited amount of syntactic information from a sentence. This project describes the identification of a specific type of noun phrase, through partial syntactic analysis, defined as a lexical noun phrase (NP_L), in texts written in Brazilian Portuguese, and annotated according to the Universal Dependency (UD) formalism. The Transformation Based Learning algorithm, TBL–Brill, applied as baseline, obtained an accuracy of 87.42% considering the UD dependency relations and 91.44% considering the UD morphosyntactic tags. Two other classifiers, one based on binary trees and the other based on a decision forest, had inferior performance.*

1. Introduction

Partial parsing, or *shallow parsing*, refers to a set of Natural Language Processing (NLP) methods aiming to retrieve a limited amount of syntactic information from a sentence. A peculiar application of *shallow parsing* that seeks to define distinct syntagmatic segments (noun phrases, verb phrases, adjective phrases, prepositional phrases, among others constituents within the text) is called *text chunking* [Hammerton et al. 2002]. Of these phrases, the noun phrases (NP) are relevant for discriminating elements with a substantive meaning and fulfill thematic roles within a sentence, encompassing functions like subject-object or agent-instrument relationships [Silva and Koch 2012].

Considering the conceptual models that categorize noun phrases, the authors Oliveira and Freitas [Oliveira and Freitas 2006] proposed the Lexical Noun Phrase (from now on NP_L), a specific NP that allows substantives, prepositions, adjectival phrases, among others, in its domains [Tjong Kim Sang and Buchholz 2000]. This type of noun phrase is critical in information retrieval and therefore is helpful for document indexing.

Currently, the Universal Dependency grammar formalism [Marneffe et al. 2021], also known as UD, is highlighted in the computational linguistics scenario. The UD is a framework for grammatical annotations across different existing natural languages. In this context, the *Center for Artificial Intelligence* (C4AI)[1], through one of

[1]https://c4ai.inova.usp.br/

its fronts, the *Natural Language Processing to Portuguese* (NLP2), seeks to enhance data and tools to enable high-level NLP of the Portuguese language, such as the Portinari project [Pardo et al. 2021], a corpus in the order of 10 million *tokens* annotated in this UD formalism. With this, in this research, we propose the identification of NP_L using morphosyntax tags (UD PoS Tag) and UD dependency relations. Due to the complexity of these phrases, the composition of rules for identifying the NP_L does not dispense a pattern recognition task through machine learning (ML) [Ramshaw and Marcus 2002]. Thus, this work increases the results obtained by [Souza and Ruiz 2022] in identifying this type of phrase.

2. Theoretical references

Syntax analysis, or parsing, is the process of analyzing and proposing an implicit grammatical structure to a sentence. Through syntactic analysis, one can determine textual patterns and understand the meanings of a sequence of terms of a logical and comprehensible structure [Jurafsky and Martin 2021]. Classical literature establishes two syntactic theories for grammatical annotation, which are: (a) constituent analysis [Chomsky 2009], and; dependency grammar [Tesnière 2015, Hjelmslev 1975]. The distinction between the two types of annotation is because the first is based on the structures of overlapping phrases; while the second is based on dependency relationships, or (in(ter))dependence[2], existing between the terms of a sentence [Pagani 2015]. Dependency grammar emphasizes the idea that linguistic units, such as words, are interconnected and interrelated.

The two theories considered in this research portray similar syntactic structures from different perspectives [Rambow 2010]. We emphasize that in the syntax of dependencies, the natural syntagmatic markings of constituents are absent and, therefore, are not made explicit. The NP_L can be characterized by presenting different syntactic lexical signatures [Souza and Ruiz 2022] and, in its extension, different gradients of complexity [Elhadad 1996].

For a brief description of NP_L one may notice that its identification, as in the example 1 below, can be trivial. However, in specific examples, whereas the NP_L are marked in bold, it is clear this identification can become a complex activity, such as we can see in the examples 2, in which the term **caneta** is a NP_L and **papel** another, and the example 3 which corresponds to a more extensive presentation of these phrases. See the examples below:

1. **A caneta** é esferográfica.
2. **Caneta** e **papel** para escrever.
3. **Caneta esferográfica Montblanc** para escrever em **papel apergaminhado de cor sépia**.

Wherever named entities, such as proper names, names of government entities or institutions, and geographic locations are encountered, they should be considered a single element. See the example 4, below, in which **João Pessoa** should be understood as a unit preserving the coordination with the term **Maceió**, that is, two distinct cores of NP_L:

[2]With the term (in(ter))dependence, we are abbreviating the three possibilities of this type of relationship: (non-reciprocal) dependence, independence (mere concatenation, without dependence on any part) and interdependence (reciprocal).

4. **João Pessoa** e **Maceió** são capitais de estados brasileiros.

The noun core restriction disregards as NP_L segments in which pronouns (example 5) and numerals play the role of central element (see example 6) because they are 'anaphoric reference to another lexical or clause element in the discourse [Oliveira and Freitas 2006].

5. **Elas** são capitais de estados brasileiros.
6. **Os três** são bons livros.

3. Related work

Ramshaw and Marcus used the Transformation-Based Learning methodology, TBL, for shallow parsing in phrase identification. The TBL method obtained precision and recall in the order of 92% for NP and 89% for other types of English phrases. Hammerton and co-authors [Hammerton et al. 2002] showed several different NLP applications that do not dispense phrasal identification. In another study on identifying noun phrases, Tham [Tham 2020] obtained an accuracy of 85.0%, and an F-measure of 85.12%. In addition to the applications on the English language, machine learning methods associated with shallow parsing were used in the Turkish language [Topsakal et al. 2017], for Hindi-English [Sharma et al. 2016], and also for Portuguese-English translation in a project developed at the University of Alicante by Garrido Alenda [Garrido Alenda et al. 2004]. Phrasal segmentation of texts written in Portuguese was also accomplished by Silva [da Silva 2007] as a shallow parser based on finite state automata. According to the researched literature, the work of Ophélie Lacroix [Lacroix 2018] introduced the identification of English NP chunks annotated under the *Universal Dependency* formalism. Following Lacroix's methodology, Souza e Ruiz [Souza and Ruiz 2022] had previously achieved an accuracy of 87.0% for the identification of NP_L and F-measures in the order of 85.3% for texts written in Portuguese in the UD context.

4. Data and methods

Data

As NP_L have their origin in the annotation noun phrases, we selected the Brazilian Portuguese Bosque *corpus* as it is annotated under two contexts: a) in the constituent grammar context, the Bosque 8.0 [Afonso et al. 2002], and; b) in the context of *Universal Dependency*, the UD_Portuguese-Bosque 2.10 [Rademaker et al. 2017]. Considering both corpora, each sentence is annotated under both theories, the constituent grammar and in the UD formalism. This way, we were able to analyze the declared syntactic structures through the hierarchical constituent model (Bosque 8.0), as well as explore their internal structures, typology, and hierarchical topology under the UD model (UD_Portuguese-Bosque 2.10). In that way, the NP_L were manually annotated in an empty field of the CoNLL-X UD file structure, a typical file structure for the *Universal Dependencies* project.

Here, we use a subset of the first 790 sentences annotated in both corpora. Table 1 describes this subset in the following fields: the number of sentences, words/*tokens*, CoNLL fields, UD Relations, and UD PoS Tags existing in the used corpus.

Table 1. The working corpus extracted from Bosque.

# Sentences	790
# Tokens	16,672
# CoNLL fields	10
# UD Relations	38
# UD PoS Tag	16
Data quantity	1020 kB

Table 2 depicts the data available to the algorithms. This table presents the terms categorized in the following fields: *token, deprel, upos, deps, IOB-format*, respectively: the corresponding word/token, the type of dependency relationship projected from the token (*deprel*), its morphosyntactic class (*upos*), its level under the dependency hierarchy (*deps*), and the marker considering the (*IOB-format*).

Table 2. Tokens and corresponding tags under UD and IOB format.

token	*deprel*	*upos*	*deps*	*IOB-format*
Averróis	root	PROPN	0	B
no	-	-	-	I
em	case	ADP	2	I
o	det	DET	3	I
poder	nmod	NOUN	1	I

Methods

Previous approaches that resorted to the use of ML for pattern identification demonstrate significant results in the application of abstract methods for composing rules using tags in the IOB format [Ramshaw and Marcus 2002]. This choice of performing certain language structures with tags that correspond to segments of interest in the text also proved to be pertinent [Ramshaw and Marcus 1999]. Considering the conceptual reflection of Santos [Santos 2021], we use algorithms that represent different mechanisms for searching sequential patterns, such as the algorithm Transformation-Based Learning, (TBL) [Brill 1995], and two classifiers, one based on decision trees and another on random forests, both using boosting, as addressed by Chen and Guestrin [Chen and Guestrin 2016].

Uneson [Uneson 2014] highlights some relevant features of the TBL algorithm, such as i) interpretability of the learned representation, ii) synthesis of the learned representation, iii) representative objective function, iv) resistance to *overtraining*, v) research during training instead of an application, vi) integration of heterogeneous resources and vii) competitive performance.

The TBL is an algorithm focused on pattern analysis that considers the positional aspect as predominant to analyze the attributes of the sentence terms. It also considers

the term's typology, order of occurrence, and place of occurrence. Its execution for composing rules considers a range, a domain, as predefined (templates). This amplitude will only be determined at the end of training. To this algorithm we present, in a first approximation, the UD relations and the markings in the IOB format corresponding to the NP_L. In a further moment, we present the UD PoS Tags with the same IOB tags. That is, we performed two experiments separately, or better, without the influence of one on the other. In the TBL methodology, the idea of learning is to start with some simple solution (initial rules) that identifies the phrases and apply transformations (new rules) that can improve the previous performance of tagging the phrases.

In the case of classifiers based on decision trees and random forests (XGBClassifier and XGBRFClassifier respectively), we emphasize that the presentation of the allowed data typology of these algorithms can be done simultaneously, that is, we can insert the fields as predictive attributes, *deprel, upos and deps*, the latter being preprocessed to extract the morphosyntactic attribute of the hierarchically superior word/*token* of the predicate/argument relation; and also the *IOB-format* tags. This way, they jointly treat independent and dependent attributes. These algorithms are characterized by identifying patterns by processing data in parallel and serial mode, that is, they search for residual patterns in features that are initially excluded by the classifier. These algorithms train under a series of trees or weak forests to obtain an increasingly robust model, in addition to having a shrinking technique that reduces the contribution of each tree in the final model, which decreases the influence of each tree, making the slower-fitting process, but resulting in robust models.

5. Results

The TBL-Brill algorithm obtained an accuracy of 87.42% through the UD relations, -4.02 p.p. below that obtained with the use of UD PoS Tags that reached 91.44% of accuracy. Thus, the TBL-Brill algorithm managed to filter representative rules with regular patterns by using only two templates, which composed six representative rules of NP_L, as summarized in Table 3.

Table 3. Final results (%) for TBL.

	Templates	Rules	Accuracy
UD dependency relations	8	57	87,42%
UD PoS Tag	2	6	91,44%

5.1. TBL rules

For the UD relations markup, we applied eight templates to compose 57 rules identified as NP_L modelers. Some of these rules are represented in Table 4. The existence of a wide range of UD tags allows for a wide range of events identified as NP_L.

As for the UD PoS tags, we noticed a high representation of the NP_L considering only two *templates* that form only the six rules represented in Table 5. Considering these rules, TBL identified 553 sequences, which represents a high performance for the different syntactic lexical signatures of NP_L.

Table 4. Some rules formed by UD relations and IOB labels.

Template	Starting tag	Final tag	Rule
017	'B'	'I'	(*token*[-1], 'det'), (*token*[1], 'flat:name')
009	'B'	'O'	(*token*[-1], 'nsubj')
017	'B'	'I'	(*token*[-1], 'case'), (*token*[1], 'flat:name')
000	'I'	'B'	(*tag*[-1], 'O')
010	'B'	'O'	(*token* [1]), 'acl:relcl')
017	'I'	'B'	(*token*[-1], 'root'), (*token*[1]),'obj')

Table 5. Main rules composed by UD PoS Tag and IOB labels.

Template	Starting tag	Final tag	Rule
017	'O'	'B'	(*token*[-1], 'ADP'), (*token*[1], 'NOUN')
017	'O'	'I'	(*token*[-1], 'DET'), (*token*[1], 'NOUN')
017	'O'	'B'	(*token*[-1], 'ADP'), (*token*[1], 'PROPN')
017	'O'	'I'	(*token*[-1], 'NUM'), (*token*[1], 'NOUN')
001	'O'	'B'	(*tag* [1]), 'I')
017	'O'	'B'	(*token*[-1], 'ADP'), (*token*[1]),'SCONJ')

One may notice that the main rules are identified by a number (either 017 or 001) and that there is a predominance of changes from initial inscription labels ('O') to final inscription labels ('B' or 'I') in patterns composed of one or more elements (*tokens*). Each tuple consists of a token at a specific position and a label corresponding to morphosyntactic markup (e.g.: 'NOUN'). In the algorithm's search for all compositions of NP_L, it found 92 possible templates, of which 18 templates were considered useful. In this specific case, template 017 obtained a score of 91.9%, that is, it was considered one of the most important templates by the model, as it composed 5 rules that represented about 83.3% of the total created rules. Meanwhile, template 001 obtained a score of 0.81%, representing lower importance concerning the other, since it established only one rule that corresponds to 16.7% of the total formed rules.

In Table 5, in its first line, we show, as an example, the sequential use of template 017, considering the tokens immediately before and immediately after the one in analysis. In composing this rule, the algorithm waits for an initial 'ADP' tag and a final 'NOUN' tag that delimits the segment corresponding to an NP_L. Analogous reasoning can be applied to the other rules. In addition, the results of applying the TBL algorithm for classification of NP_L, shown in Table 6, express the metrics of precision, recall, F-measure, and accuracy for the IOB markings conditioned to the POS labels. We see that comparing the TBL performance between the UD dependency ratios and the POS markings, the latter obtained slightly more advantageous results, culminating in an accuracy of 91.4% when considering the POS and 87.4% considering only the UD relations.

Table 6. TBL comparative percentage results for IOB tags.

Tags	Metrics			
	Precision	Recall	F-measure	Accuracy
UD relations				87,42
B	78,66	88,02	83,08	–
I	88,79	76,34	82,09	–
O	88,94	89,47	89,20	–
UD PoS Tag				91,44
B	92,18	93,81	92,99	–
I	90,91	85,89	88,33	–
O	91,05	92,91	91,97	–

5.2. Classifiers based on decision trees and forests with boosting

Remember that we also tested two other classifiers, which are the XGBClassifier and the XGBRFClassifier. The XGBClassifier is a machine learning algorithm applied to structured data. This classifier is an implementation of decision trees with gradient boosting. This implementation is focused on performance gain. XGBoost is the basis of the XGBClassifier classifier and is typically used to train gradient-boosting decision trees. XGBRFClassifier is a version of XGBClassifier trained using random forest.

The XGBClassifier algorithm obtained a precision of 87.19%, that is, a value of only +0.86 p.p. above its peer based on decision forest. Similarly, we found an increment of +0.58 p.p. on revocation, also +0.62 p.p. in the F-measure, and +0.58 p.p. for accuracy. Modest but superior increments.

Following its experiment, we applied another closed cross-validation method. The results showed a mean accuracy of 86.78% for XGBClassifier, a +0.13 p.p increase compared to its previous result; and 86.18% for XGBRFClassifier, a +0.11 p.p. increase when compared to its previous result.

Table 7. Results (%) of classifiers based on decision trees and forests.

Classifiers	Metrics			
	Precision	Recall	F-measure	Accuracy
XGBClassifier	87,19	86,65	86,78	86,65
XGBRFClassifier	86,33	86,07	86,16	86,07

6. Conclusion

Partial parsing proved to be a plausible strategy for the identification of NP_L, in texts written in Portuguese and annotated in the UD formalism, through machine learning tech-

niques and rule abstraction, the latter implemented using IOB tags. Computational learning that resorts to the classification of IOB labels allows the identification of fragments that compose an NP_L and, therefore, its more extensive configurations may have their limits poorly defined or discontinued.

As for the computational treatment with the TBL algorithm, the UD PoS Tags stood out with a percentage of +4.02 p.p. in accuracy (91.44%) when comparing the marks of UD Dependency Relations (87.42%). Considering the other algorithms, those based on decision trees and forests, they obtained an accuracy metric of at least 4.79 p.p. more expressive (86.65% for the XGBClassifier). The cross-validation technique slightly improved the application, achieving +0.13 and +0.11 mean accuracy. The performance of TBL in these tests for the Brazilian Portuguese language does not represent an appropriate result to state that ML applications may have similar performance for other natural languages annotated in the UD formalism since the TBL algorithm, when establishing a sequence for identifying the NP_L, is subordinated to the specific typology of terms in this other language. We also point out that the dependence of many current algorithms on the volume and variety of data for pattern extraction can influence the results.

The research carried out by Oliveira and Freitas [Oliveira and Freitas 2006] to identify NP_L is inserted in another syntactic context, the context of constituents. These researchers brought a new relevant syntagmatic specification to the computational linguistics scenario, the NP_L, and reached an accuracy of 85.9% and an F-measure of 86.2%. Establishing a comparison between this and the research by Oliveira and Freitas would be inappropriate since they treat different data annotated under different grammatical formalisms. However, the proximity between the metrics obtained in these two experiments confirms Rambow's assertion that constituent and dependency grammars bring the same syntactic content from different perspectives [Rambow 2010]. We emphasize that UD morphosyntax, combined with UD dependency relations, are elements that allow the establishment of non-natural syntagmatic segments of the universal dependency grammar.

Finally, we highlight possible future contributions: i) the expansion of the *corpus* with a greater number of annotated sentences to reaffirm or not TBL's performance against such state-of-the-art algorithms; ii) approximations that increase the accuracy and recall achieved so far; iii) the identification of this specific type of phrase in other languages to reaffirm the proposal of the Universal Dependency project, as well as the correlation of the NP_L to other natural languages, and; iv) verify if only with the restricted use of UD dependency relations in another natural language the typological question can be crossed.

Acknowledgement

This work was carried out at the Center for Artificial Intelligence of the University of São Paulo (C4AI – `http://c4ai.inova.usp.br/`), with support by the São Paulo Research Foundation (FAPESP grant #2019/07665-4) and by the IBM Corporation. The project was also supported by the Ministry of Science, Technology and Innovation, with resources of Law N. 8.248, of October 23rd, 1991, within the scope of PPI-SOFTEX, coordinated by Softex and published as Residence in TIC 13, DOU 01245.010222/2022-44.

References

Afonso, S., Bick, E., Haber, R., and Santos, D. (2002). Floresta sintá(c)tica: A treebank for Portuguese. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC)*, pages 1698–1703, Las Palmas, Spain.

Brill, E. (1995). Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. *Comput. Linguist.*, 21(4):543–565.

Chen, T. and Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD'16, page 785–794, New York, NY, USA. Association for Computing Machinery.

Chomsky, N. (2009). *Syntactic structures*. De Gruyter Mouton.

da Silva, J. R. M. F. (2007). *Shallow processing of Portuguese: From sentence chunking to nominal lemmatization*. PhD thesis, Universidade de Lisboa, Faculdade de Ciências.

Elhadad, M. (1996). Lexical choice for complex noun phrases: Structure, modifiers, and determiners. *Machine Translation*, 11:159–184.

Garrido Alenda, A., Gilabert Zarco, P., Pérez-Ortiz, J. A., Pertusa, A., Ramírez Sánchez, G., Sánchez-Martínez, F., Scalco, M. A., and Forcada, M. L. (2004). Shallow parsing for Portuguese-Spanish machine translation. In *Workshop Notes of TASHA'2003*, pages 21–24, Lisboa, Portugal. Edições Colibri.

Hammerton, J., Osborne, M., Armstrong, S., and Daelemans, W. (2002). Introduction to Special Issue on Machine Learning Approaches to Shallow Parsing. *Journal of Machine Learning Research*, 2:551–558.

Hjelmslev, L. (1975). *Prolegômenos a uma teoria da linguagem*. Perspectiva.

Jurafsky, D. and Martin, J. (2021). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, volume 3. Stanford Edu.

Lacroix, O. (2018). Investigating NP-Chunking with Universal Dependencies for English. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 85–90, Brussels, Belgium. Association for Computational Linguistics.

Marneffe, M.-C., Manning, C. D., Nivre, J., and Zeman, D. (2021). Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Oliveira, C. and Freitas, M. C. (2006). Um modelo de sintagma nominal lexical na recuperação de informações. *XI Simpósio Nacional e I Simpósio Internacional de Letras e Linguística (XI SILEL)*, pages 778–786.

Pagani, L. A. (2015). Duas Noções Fundamentais para Gramáticas de Dependência.

Pardo, T., Duran, M., Lopes, L., Felippo, A., Roman, N., and Nunes, M. (2021). Porttinari – a Large Multi-genre Treebank for Brazilian Portuguese. In *Anais do XIII Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, pages 1–10, Porto Alegre, RS, Brasil. SBC.

Rademaker, A., Chalub, F., Real, L., Freitas, C., Bick, E., and de Paiva, V. (2017). Universal Dependencies for Portuguese. In *Proceedings of the Fourth International Confer-*

ence on Dependency Linguistics (Depling 2017), pages 197–206, Pisa, Italy. Linköping University Electronic Press.

Rambow, O. (2010). The Simple Truth about Dependency and Phrase Structure Representations: An Opinion Piece. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 337–340, Los Angeles, California. Association for Computational Linguistics.

Ramshaw, L. and Marcus, M. (2002). Text Chunking Using Transformation-Based Learning. *Third ACL Workshop on Very Large Corpora. MIT*, pages 157–176.

Ramshaw, L. A. and Marcus, M. P. (1999). Text chunking using transformation-based learning. *Natural language processing using very large corpora*, pages 157–176.

Santos, D. S. M. (2021). Grandes quantidades de informação: um olhar crítico. In *II Congresso Internacional em Humanidades Digitais*, Online. UFRJ.

Sharma, A., Gupta, S., Motlani, R., Bansal, P., Shrivastava, M., Mamidi, R., and Sharma, D. M. (2016). Shallow Parsing Pipeline – Hindi-English Code-Mixed Social Media Text. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1340–1345, San Diego, California. Association for Computational Linguistics.

Silva, M. C. and Koch, I. G. (2012). *Linguística aplicada ao português.* Cortez.

Souza, A. and Ruiz, E. E. S. (2022). Investigating Lexical NP-Chunking with Universal Dependencies for Portuguese. In *Anais do XIX Encontro Nacional de Inteligência Artificial e Computacional*, pages 342–351, Porto Alegre, RS, Brasil. SBC.

Tesnière, L. (2015). *Elements of structural syntax*. John Benjamins Publishing Company.

Tham, M. J. (2020). Bidirectional Gated Recurrent Unit For Shallow Parsing. *Indian Journal of Computer Science and Engineering (IJCSE)*, 11(5):517–521.

Tjong Kim Sang, E. F. and Buchholz, S. (2000). Introduction to the CoNLL-2000 Shared Task Chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*.

Topsakal, O., Açikgöz, O., Gürkan, A. T., Kanburoglu, A. B., Ertopçu, B., Özenç, B., Çam, I., Avar, B., Ercan, G., and Yildiz, O. T. (2017). Shallow parsing in Turkish. In *2017 International Conference on Computer Science and Engineering (UBMK)*, pages 480–485.

Uneson, M. (2014). When Errors Become the Rule: Twenty Years with Transformation-Based Learning. *ACM Comput. Surv.*, 46(4).