Rule-based Surface Realization of Roma

Ciprian-Virgil Gerstenberger
UiT The Arctic University of Norway
ciprian.gerstenberger@gmail.com

Abstract

Due to its reliance on context and intricate grammatical rules, the Romanian weak pronoun system presents a challenge not only for language learners – both native and non-native speakers – but also for linguistic description and computational processing.

The present work addresses the challenges of Romanian weak pronouns from a computational processing perspective. Accordingly, it has three main goals: (1) to present the implementation of a rule-based model for generating contextually accurate surface forms of Romanian weak pronouns, (2) to describe the compilation of a database of relevant inputs for testing surface realization, and (3) to test the efectiveness of the model.

This serves as a proof of concept demonstrating both the transparency and the efectiveness of the model when based on an appropriate linguistic description.

1 Introduction

Romanian weak pronouns (henceforth, RWPs), commonly referred to as clitics, exhibit a variety of surface forms governed by intricate, contextually dependent morpho-phonological rules. As such, RWPs pose challenges not only for linguistic description but also for computational modelling and natural language processing (henceforth, NLP).

Despite extensive research on Romance clitics within various theoretical frameworks – including Generative Grammar, as explored by Dobrovie-Sorin (1999), Some (2007), and Săvescu Ciucivara
cal Functional Grammar (LFG),
by Barbu and Toivonen (2018);
Phrase Structure Grammar (HP
cussed in Monachesi (2001) an
(2005); various Optimality Th
dels, advanced by Popescu (200
Căluianu (2000), Legendre (2
(2003), and Cherecheș (2014); and Dynam
Syntax, as promoted by Klein (
remains a need for practical,
tools that can accurately gene
their usage in real-world con

The current study addresses
cusing on the creation of a ru
designed to handle the surfac
RWPs, tested using a speciali
base. This, in turn, contribu
der goal of enhancing computat
applications for Romanian.

2 Romanian Weak Pronouns

As with other Romance languag
employs a pronominal system t
both strong pronouns and weak
nouns. Romanian has a complex
with fully marked case, numbe
distinctions, comprising bot
flexive RWPs. Moreover, there
main weak verb (henceforth
the to behave similar
RWPs can appear in both s
asyllabic forms. The syllab
well as their requirement for as
gatory and hence) for the possibil
(optional sandhi), is determi
context of occurrence. Sandhi r
logical adjustments that occ
word boundaries, infuenced
tors. The syllabic (and thus

forms are referred to as sandhi forms.

RWPs occur in a fixed order of up to three elements [RWP $\underline{dat\_ethicus}$ < RWP $\underline{dat}$ < RWP $\underline{acc}$] in clitic clusters (or clitic sequences), alongside other clitics such as negation, auxiliary verbs, or adverbial particles (ex. 1). The cluster can occur both in preverbal position (e.g., in declarative sentences, ex. 2) and postverbal position (e.g., in imperative sentences, ex. 3), with the exception of negation and adverbial particles, which can occur only preverbally.

(1) Nu ni le-ai mai dat.
not cl_1.pl.dat cl_3.pl.acc have_2.sg.pres more given
«You didn't give them to us anymore.»

(2) Mi le dai acum.
cl_1.sg.dat cl_3.pl.acc give_2.sg.pres now
«You give them to me now.»

(3) Dă-mi-le acum!
give_2.sg.imp cl_1.sg.dat cl_3.pl.acc now
«Give them to me now!»

(4) Le dai mere.
cl_3.pl.dat give_2.sg.pres apple_pl.acc
«You give apples to them.»

Romanian exhibits various types of ambiguity that complicate the interpretation of the RWP data, such as case syncretism between accusative (ex. 2) and dative (ex. 4) plural, part-of-speech homonymy (between the dative-reflexive RWP își and the conjunction și «and» in ex. 5), phoneme-grapheme ambiguity (syllabic form [mi] in ex. 6 vs. asyllabic form with a glide [mʲ] in ex. 7), as well as hyphen ambiguity (marking asyllabicity in mi-o in ex. 7 vs. marking postverbality in dă-mi-le in ex. 3).

(5) Și le cumpără
cl_3.sg.dat cl_3.pl.acc buy_3.sg.pres
și și le
and_conj cl_3.sg.dat cl_3.pl.acc
revinde.
resell_3.sg.pres
«He/she buys them for him-/herself and resells them for him-/herself»

(6) Mi-l dai.
cl_1.sg.dat cl_3.m.sg.acc give_2.sg.pres
«You give it to me.»

(7) Mi-o dai.
cl_1.sg.dat cl_3.f.sg.acc give_2.sg.pres
«You give her/it to me.»

(8) Îmi dai cartea.
cl_1.sg.dat give_2.sg.pres book_def.sg.acc
«You give me the book.»

(9) Dă-mi cartea!
give_2.sg.imp cl_1.sg.dat book_def.sg.acc
«Give me the book!»

As a general observation, syl
whether obligatory or optiona
in the rightmost RWP. Obligat
the following item, to the ri
by the occurrence of an auxili
with a vowel (e.g., am, ai, aș) or the
3.sg. acc. form o (mi-o in ex. 7).

Obligatory sandhi to the pre
the left, occurs if the conte
sandhi to the right is not pres
derlying form (see Section 3)
item ends in u. If the preceding i
also an RWP it serves as syllab
rightmost RWP (mi-l in ex. 6). If the r
most RWP is the only item in the
surface as prosthetic form in preve
tion (îmi in ex. 8). In postverbal p
verb functions as the syllabi
(dă-mi in ex. 9).

The RWP underlying forms ni, vi
(1./2.pl.dat) exhibit a special behavio
as single RWPs, thus in both t
and leftmost position, they su
cusative counterparts ne, vă respecti-
vely – an instance of the afore
syncretism (ex. 4). In other clu
sitions, the surface forms re
the underlying forms (ex. 1).

When obligatory sandhi is no
specific contextual condition
nian RWPs can undergo optional
means that they may reduce (asy
11), but such a reduction is n
(syllabic le in ex. 10). Optional san
following item, to the right,
following item starts with an
(ex. 11). Similarly, some RWP
of length one can optionally a
ceding item, to the left, if th
ends with an unstressed vowel (să-i in ex.
12). In the same context, the
surface as an î-prosthetic for (îmi in ex. 13).

(10) Le aduci mere
cl_3.pl.dat bring_2.sg.pres apple_pl.acc
«You bring them apples.»

(11) Le-aduci mere
cl_3.pl.dat bring_2.sg.pres apple_pl.acc

«You bring them apples.»

(12) Vreau să-mi dai mere.
want_1.sg that cl_1.sg.dat give_2.sg apples
«I want you to give me apples.»

(13) Vreau să îmi dai mere.
want_1.sg that cl_1.sg.dat give_2.sg apples
«I want you to give me apples.»

3 Model description

The computational implementation of a theoretical model serves as a bridge between abstract concepts and practical applications, providing a platform to explore, validate, and extend the capabilities of the model. In linguistics, where numerous frameworks come with diverse theoretical constructs, translating these abstract models into computational terms becomes essential (see, for instance, Bender and Langendoen, 2010).

Although numerous computational tools for linguistics are available – such as Hayes et al. (2013) for OT, Kaplan et al. (2004) for LFG, Copestake (2001) for HPSG, to name a few –, none of the theoretical approaches to RWPs mentioned in Section 1 have attempted to demonstrate how these frameworks can be computationally implemented and systematically validated.

In Gerstenberger (2022), I provided comprehensive description of RWPs and their contexts of occurrence using transparently intersubjectively testable linguistics features based on their (a)syllabicity. Granted, the syllable is notoriously difficult to define in a universally agreed-upon way – is it primarily a unit of speech production (articulation), perception, or both? (cf., for instance, Ohala 2008), but it is widely recognized as a fundamental building block in phonological theory.

Unlike Dobrovie-Sorin and Giurgea's (2013: p. 266) claim that „clitic forms are underlyingly asyllabic or syllabic”, Popescu's (2003: p. 154) claim of an unspecified underlying mora, or Klein's (2007: p. 77) use of cluster-prosthetic forms as model input, I propose a model in which all underlying representations are uniformly syllabic (see Table 1).

The model I propose for handl
similar to the generative app
sky and Halle (1968) and the t
phology in Koskenniemi (1983)
two different levels of represe
a set of rules for mapping betw
lying and surface levels.

While the underlying repres
theoretical linguistic entit
are their computational ling
parts – concrete strings that

Given the case syncretism be
tive and dative plural RWP for
in Section 2 (ne ⇒ n, −syllabic) and the
(syll ⇒ asyllabic), as well a
observation that the obligato
right has higher precedence t
tory sandhi to the left, a cons
as presented for XFST in Beesl
nen (2003), seems better suit
these rules.

The key idea when modeling RW
mena is to account for the pos
in the clitic sequence: is the
most position or not? Next, it
identify the correct contextu
tinguishing between obligato
sandhi. The asyllabic RWP form
by vowel loss, hyphen addition
the postverbal position is cha
addition of hyphens between a
clitic sequence that follow t

The following XFST grammar
provides definitions and rules
obligatory sandhi to the righ
that are followed by an auxili
with a vowel or by the RWP form

```
define V_RWP "o";
define ReduceHighV_RWP "mi" | "ți" |
    "și";
define DeleteHighV_RWP "u";
define SubstituteHighV_RWP "ni" | "
define HighV_RWP ReduceHighV_RWP |
    DeleteHighV_RWP | SubstitueHigh
define DeleteLowV_RWP "mă" | "se" |
define LowV_RWP DeleteLowV_RWP | "
    "le";
define Syllabic_RWP HighV_RWP | Lo
define V_Initial_Aux "am" | "ai" |
define C_Initial_Aux "voi" | "vei"
    "vom" | ...;
define Aux V_Initial_Aux | C_Initi
define IsObligatoryHost V_Initi
    V_RWP;
define RWP V_RWP | Syllabic_RWP;
define Rightmost_RWP Syllabic_RWP
```

Number	Accusative					Dative af				Ve
	1p	2p	3p.m	3p.f	3p.ref	1p	2p	3p.m	3p.f	3p
Sg	/mə/	/te/	/lu/	/o/	/se/	/mi/	/tsi/	/i/	/i/	/i
Pl	/ne/	/və/	/i/	/le/	/se/	/ni/	/vi/	/li/	/l	

Tabela 1: Underlying forms as input for the surface fo «to be»

```
   Syllabic_RWP .#. Syllabic_RWP;
define Remove_Vowel
   (DeleteLowV_RWP | DeleteHighV_RWP) -> (""
   / [aeiouă] _ )
   / .#. Is_Obligatory_Host;
define Substitute_Form
   "ni" -> "ne" / _ Rightmost_RWP .#. ||
   "vi" -> "vă" / _ Rightmost_RWP .#. ||
   "li" -> "le" / _ Rightmost_RWP .#. ;
define Add_Right_Hyphen Rightmost_RWP -> "-" /
    _ Is_Obligatory_Host;
define Asyllabic_Transformation Remove_Vowel
   || Substitute_Form || Add_Right_Hyphen;
regex Asyllabic_Transformation;
```

First, groups of items with the same behavior are defined: ReducedHighV_RWP which have the same orthographic string in both syllabic and asyllabic forms, DeleteHighV_RWP and DeleteLowV_RWP lose the vowel as asyllabic forms, as encoded in Remove_Vowel. HighV_RWP items change they form according to Substitute_Form, reflecting the merged case syncretism.

The positional information – whether an RWP is the rightmost in the sequence – is modeled by Rightmost_RWP: a rightmost RWP is not followed by any other RWP, which is expressed using set subtraction.

In the context of the obligatory host to the right, Add_Right_Hyphen ensures that the correct orthographic forms are produced as surface forms.

4 Model implementation

While the XFST grammar fragment in the previous section is a proof that the linguistic phenomena described in Section 2 can be translated into XFST rules, the concrete implementation has been done in Python. There are some non-negligeable reasons for Python:

- Python allows for more expressive rule implementations.
- Python has excellent debugging tools.
- Python has a rich ecosystem as spaCy, NLTK, Stanza.
- Python syntax is cleaner an read than XFST syntax.
- Python scripts can be easily shared, deployed, or packag

The task of the surface form g is to produce the correct RWP within its specific contex prior to the syllabi The following operation deleting a vowel, changing a v and marking the hyphen, adding adding postverbal hyphens The module expects the input necessary linguistic informa in a given context. During the process, the procedure first chec each item and applies the cons gatory sandhi to the right, t followed by those on optional ditional surface form adjustm rund and imperative forms may context. If no sandhi constra llabic underlying RWP forms r ged.

Since the main task of this p proof of concept – namely, to that the linguistic description Stenberger (2022) can be impl spaCy[1], an open-source soft brary for advanced NLP in Pytho for this task. spaCy Doc objects contain default linguistic an part-of-speech tags and posit of specific tokens in context, tic structure can also be ada

[1] https://spacy.io/

and customized for specific types of tion. When run on the input, can be queried and checked lues, allowing operations each token.

The input sentence (see is analyzed using spaCy created specifically for this purpose. As shown in the code fragment in Figure 1, the Python rules operate on each token in the spaCy Doc object.

For instance, the code checks whether the current token is a clitic pronoun, as defined in Gerstenberger (2007), meaning that it can occur in both preverbal and postverbal positions. If this condition is met, it further checks whether the token is an RWP or an RWV to determine whether additional operations should be performed on the input string. Along with default value checks on the current token – such as its current position and contextual elements – custom functions like token._.is_rwp, token._.is_rwv and is_rightmost_in_cluster(token) have been implemented for this task. To address the dative/accusative case syncretism in the plural, term mapping has been employed. The entire rule set is freely available from https://github.com/ciprian-NO/rb_rwp_generation/blob/main/generate_rwp_surface_forms.ipynb.

To test the effectiveness of the module, I created a test input database covering all relevant RWP phenomena described in Section 2.

5 Test database creation

Evidently, the design of the test input database is driven by the RWP surface realization model proposed. The structure of a database entry is a tuple <INPUT, {TARGET$_1$, ..., TARGET$_n$}>, where the input contains the syllabic input representation of RWPs and the target contains all possible output variants. For example, for the current input s-ar toate («May all his/her bones stiffen!»), the input is [s-ar toate oase], created by removing the hyphens that indicate post-syllabicity. Furthermore, the target

surface form may be replaced by t syllabic and Due to grapheme phoneme ambiguities in RWPs 2 more because both for a syllabic and an asyl

A segment (see Section vis sandhi refers to phonolog at morpheme or word boundaries influenced by factors such as idiolect), soc lects (sociolect, regiolect) speech register Our output targets, these are bas minimal that that can o context.

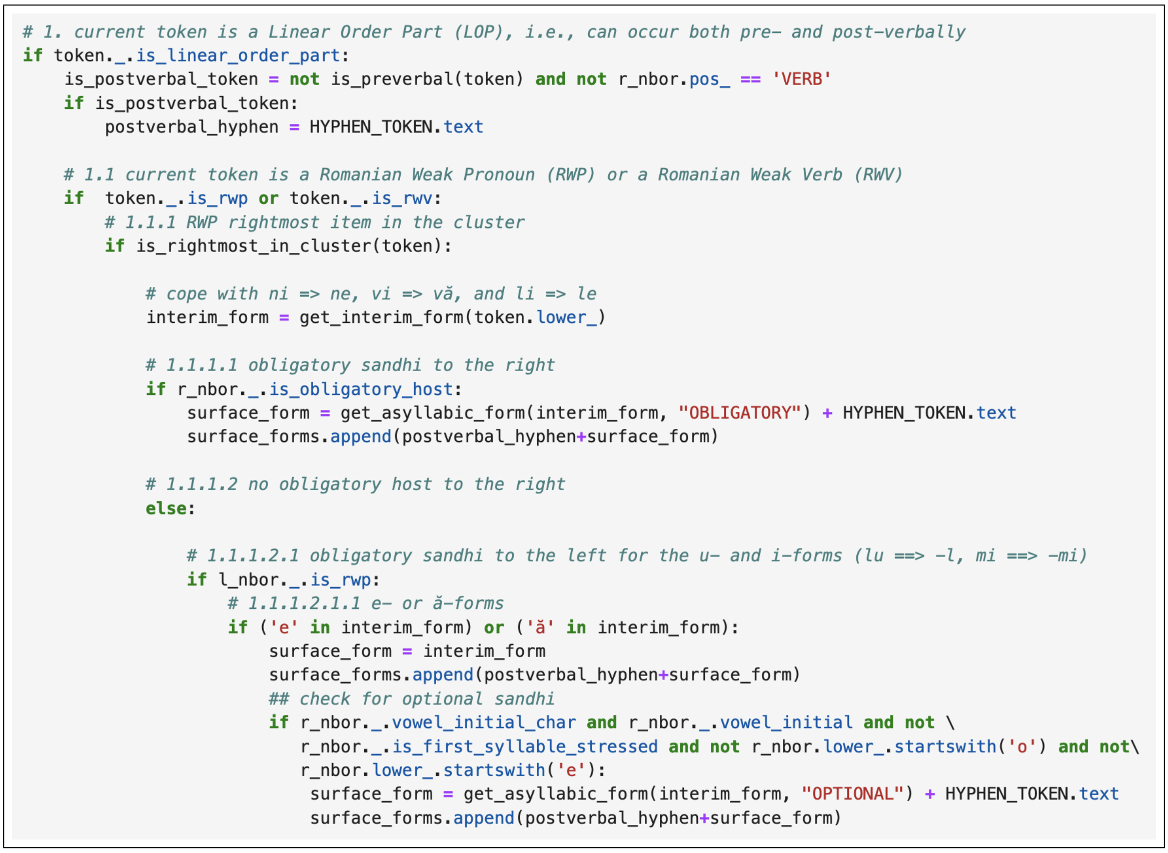
Figure 2 shows a database ent one target. In such cases, t is identical Arată-ne muzeu («Show us the museum!»): bo the correct surface form of th

In database entries that co targets reflecting optional s sentence: «Unde ești?» («Where are you giving him/her apples?») in împușcă în inimă («He/she shoots her heart.») in Figure 4, the only one of the different targets.

RWP configurations are compl their syntactic positioning, havior, and interactions wit in the sentence. When buildin test input database for testi alization of RWPs, several ke structural factors (see Secti sidered, such as: person, num case of RWPs, cluster length,

The test input database has b created based on examples fro (2022) and the relevant l therein. RWP examples that we plete sentences have been mini into full sentences (the da- tabase entry) typically by a (e.g. cumpără-l («Buy it!») → Cumpără-l repede! («Buy it quickly!»). Some R expressions are repeated in differen surface realization in cases

The database contains 352 inp 470 target sentences, distri 257 have a single target outpu two targets, 17 sets wi



```
# 1. current token is a Linear Order Part (LOP), i.e., can occur both pre- and post-verbally
if token._.is_linear_order_part:
    is_postverbal_token = not is_preverbal(token) and not r_nbor.pos_ == 'VERB'
    if is_postverbal_token:
        postverbal_hyphen = HYPHEN_TOKEN.text

    # 1.1 current token is a Romanian Weak Pronoun (RWP) or a Romanian Weak Verb (RWV)
    if  token._.is_rwp or token._.is_rwv:
        # 1.1.1 RWP rightmost item in the cluster
        if is_rightmost_in_cluster(token):

            # cope with ni => ne, vi => vă, and li => le
            interim_form = get_interim_form(token.lower_)

            # 1.1.1.1 obligatory sandhi to the right
            if r_nbor._.is_obligatory_host:
                surface_form = get_asyllabic_form(interim_form, "OBLIGATORY") + HYPHEN_TOKEN.text
                surface_forms.append(postverbal_hyphen+surface_form)

            # 1.1.1.2 no obligatory host to the right
            else:

                # 1.1.1.2.1 obligatory sandhi to the left for the u- and i-forms (lu ==> -l, mi ==> -mi)
                if l_nbor._.is_rwp:
                    # 1.1.1.2.1.1 e- or ă-forms
                    if ('e' in interim_form) or ('ă' in interim_form):
                        surface_form = interim_form
                        surface_forms.append(postverbal_hyphen+surface_form)
                        ## check for optional sandhi
                        if r_nbor._.vowel_initial_char and r_nbor._.vowel_initial and not \
                           r_nbor._.is_first_syllable_stressed and not r_nbor.lower_.startswith('o') and not\
                           r_nbor.lower_.startswith('e'):
                            surface_form = get_asyllabic_form(interim_form, "OPTIONAL") + HYPHEN_TOKEN.text
                            surface_forms.append(postverbal_hyphen+surface_form)
```

Figura 1: Example of RWP surface generation rules

and 8 sets with four. The main features of the RWPs are summarized in Table 2. Even if a database with only 352 input strings might seem small, it contains all relevant phenomena for handling with RWPs.

Both the input string and target sets were manually created. The input string is generated by replacing all instances of RWP surface forms with their underlying forms and removing any hyphen in the example sentence, representing an abstraction process. In contrast, creating the target sets involves generating all possible variants of the input string, representing an instantiation.

For contexts involving RWPs, all possible combinations have been created. However, in cases of optional sandhi without RWPs, only those variants covered by the rules implemented for RWP contexts have been generated. Moreover, since there is no consensus on what precisely triggers the selection of variants in optional sandhi, the decision was made to include all potential variants, even though some may be more acceptable than others depending on the context.

When selecting examples for
tables, we avoided using RWP se
widely accepted, such
RWPs (see Dobrovie-Sorin and G
p. 262) or examples of RWP cli
involving marginally accepta
Constraint (PCC), a constrai
the co-occurrence of certain
specifically based on two featu
person case (cf. Bonet, 1994).

While not aiming to be exhaust
RWP combinations, we sought to
the phenomena described in
(2020). More precisely, this
ated manually and has been inc
rected. It is not immune to er
linguistic judgment or spelli

As a final remark, the same inp
base is intended for use in tes
based methods, mo
Pre-Trained Transfo
Given the widespread popula
variability, we decided
large language models provid
(2024). The goal of using a s

Feature	Preverbal forms	Postverbal forms	Example
RWP sequence length	L1 280 L2 79 L3 16	L1 71 L2 31 L3 7	Postverbal L2 Dă-mi-le acum! «Give them to me now!»
	Obligatory î-prothetic forms	Optional î-prothetic forms	
Î-prothetic contexts (Only preverbal L1)	25	î-prothetic forms: 43 no î-prothetic forms: 50	Optional î-prothetic form De ce îmi dai mere? «Why are you giving me apples?»
	Obligatory forms	Optional sandhi contexts	
No î-prothetic contexts	263	RWP as syllabic host: 32 Context item as syllabic host: 50	Obligatory RWP forms De ce mi le dai? «Why are you giving them to me?»
	Syllabic context items	Asyllabic context items	
Optional sandhi contexts with RWP as possible syllabic host	15	17	RWP as syllabic host N-o văd bine. «I don't see her/it well.»
	Syllabic RWP forms	Asyllabic RWP forms	
Optional sandhi contexts with context item as possible syllabic host	25	25	Context item as syllabic host El ne-arată muzeul. «He shows us the museum.»

Tabela 2: Distribution of RWP features of target s

put database is to establish a fair comparison and evaluation between rule-based and GPT-based approaches.

6 Feature annotation and validation

The model relies on fine-grained linguistic information about both the currently processed item and its context, which necessitates that the input data be linguistically annotated accordingly.

As already mentioned, for processing the entire set of 352 input sentences, I chose to use spaCy for several reasons:

- it offers multi-language support (for over 75 languages, including Romanian[2]);
- it offers a hybrid approach, i.e., it supports both machine learning models and linguistic rules;
- it is highly customizable and extensible (e.g., via Custom Token Attributes such as is_obligatory_host).

To address both obligatory and optional sandhi in RWP surface realization, the spaCy annotations have been extended with a set of functions that enrich token annotation with features for grapheme-phoneme disambiguation (such as vowel_final) and indicate whether an initial vowel is stressed (is_first_syllable_stressed).

Another set of functions has been devised to add high-level annotations, such as is_rwp, is_rightmost_in_cluster

[illegible] These functions
[illegible] morphosyntactic a
are designed to simplify the v
textual features for generati
RWP form.

Finally, to ensure that hyph
asyllabicity, postverbality
ned according to official writi
the annotations order_par
is_preverbal have been added to the c
spaCy token features. In contras
position (ex. 2), all items in
ding RWPs and auxiliary verbs
ted to the main verb by hyphens
position (ex. 3).

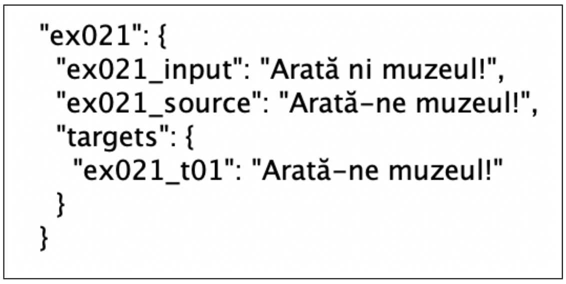
Each input string is annota
spaCy module, extended with func
ditional features, and the res
in JSON format. Task-relevan
are then manually corrected,
JSON structures are converted to spaCy bi-
nary format. Since only task-
tations are manually correcte
tions may contain errors.

7 Output testing

As with the development of any
crafted grammar, the impleme
RWP surface generation module
tive process involving carefu
finement, and extensive testin
linguistic patterns were acc
Each iteration required revis
les, adding new ones, and addr
ses.

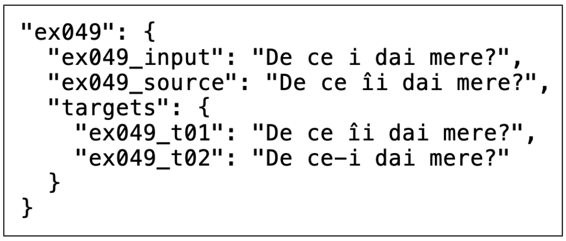
The spaCy-annotated, manual

[2] https://spacy.io/models/ro



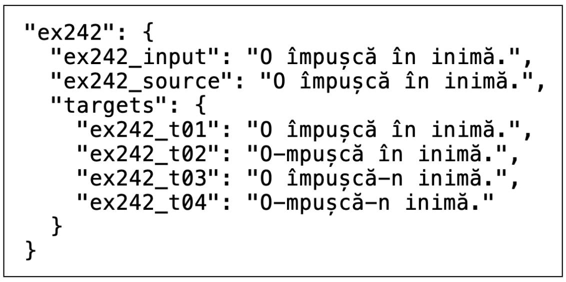
```
"ex021": {
  "ex021_input": "Arată ni muzeul!",
  "ex021_source": "Arată-ne muzeul!",
  "targets": {
    "ex021_t01": "Arată-ne muzeul!"
  }
}
```

Figura 2: Database entry with one target



```
"ex049": {
  "ex049_input": "De ce i dai mere?",
  "ex049_source": "De ce îi dai mere?",
  "targets": {
    "ex049_t01": "De ce îi dai mere?",
    "ex049_t02": "De ce-i dai mere?"
  }
}
```

Figura 3: Database entry



```
"ex242": {
  "ex242_input": "O împușcă în inimă.",
  "ex242_source": "O împușcă în inimă.",
  "targets": {
    "ex242_t01": "O împușcă în inimă.",
    "ex242_t02": "O-mpușcă în inimă.",
    "ex242_t03": "O împușcă-n inimă.",
    "ex242_t04": "O-mpușcă-n inimă."
  }
}
```

Figura 4: Database entry with four targets

mini-corpus based on the input data is loaded and used to re-analyze the input so that the linguistic features needed for RWP surface generation are available for the application of the rules. Finally, the output is checked against the corresponding set of targets (see Figures 2–4).

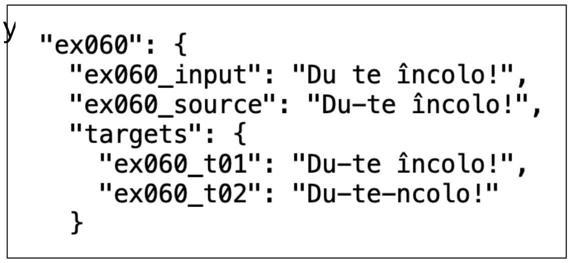
Since this implementation is primarily a proof of concept – for immediate context checking –, I kept the processing as simple as possible. This simplicity means that the module has a stateless design, i.e. no «memory»: it does not retain information about the previously generated form of the current token nor of the previously generated neighboring token. In standard Romanian orthography, hyphens can denote sandhi, postverbal clitics, or both. Given the lack of memory in the module, a hyphen may occasionally be generated twice.

For each token, a set of surface forms is generated based on its context. The resulting ou

tput strings then comprises a
binations of these token surf
of overgeneration are filtered
step of processing, as illust
and 4. below.

```
<060> . . . . . . . . . . . . . . . .
[4 tokens] Du te încolo!
[['Du'], ['-te', '-te-'],
 ['încolo', '-ncolo'], ['!']]
 1. ('Du', '-te', 'încolo', '!')
 2. ('Du', '-te', '-ncolo', '!')
 3. ('Du', '-te-', 'încolo', '!')
 4. ('Du', '-te-', '-ncolo', '!')
. . . . . . . . . . . . . . . . . . .
```

The remaining correct output Du-
te încolo! and Du-te-ncolo! («Go away!»)
are checked against the targe
ponding database entry in Fig



```
"ex060": {
  "ex060_input": "Du te încolo!",
  "ex060_source": "Du-te încolo!",
  "targets": {
    "ex060_t01": "Du-te încolo!",
    "ex060_t02": "Du-te-ncolo!"
  }
}
```

Figura 5: Entry with optiona
gets

The hand-crafted grammar is
cifically for a defined input se
cular output set, making it n
in its linguistic choice of ex
tions nor entirely error-free
or implementation. Neverthel
an attempt to faithfully impl
description outlined in Gers
within a computational-lingu

8 Conclusions

In this article, I presented t
of a model for RWP surface rea
lowing the linguistic descri
Gerstenberger (2022). While
(2018) addresses only the rea
gatory sandhi and does not of
tional implementation of the
study extends the approach by
both obligatory and optional
computational framework.

A further key contribution o
the compilation of a systemat
The evaluation of the model

linguistic annotations. Beyond the implemented constraints, this computational implementation serves not only as a proof of concept but also as a robust method for validating the model, demonstrating its practical applicability, and ensuring that its predictions align with empirical data. Due to its transparency and proximity to linguistic surface forms, the model is adaptable to any constraint-based linguistic framework.

Since the resources used in this study—specifically, the input test database and the surface generation module—were created manually, they may contain errors. However, these resources are freely available at https://github.com/ciprian-NO/rb[illegible]_generation for the research community to test, improve, and expand.

References

Roxana-Maria Barbu and Ida Toivonen. 2018. Romanian Object Clitics: Grammaticalization, agreement and lexical splits. In Proceedings of the LFG18 Conference, pages 67–87. Stanford: CSLI Publications.

Kenneth R. Beesley and Lauri Karttunen. 2003. Finite-State Morphology. CSLI Publications, Stanford, CA.

Emily M. Bender and D. Terence Langendoen. 2010. Computational linguistics in support of linguistic theory. Linguistic Issues in Language Technology, 3.

Eulàlia Bonet. 1994. The person-case constraint: A morphological approach. MIT Working Papers in Linguistics, (22):33–52.

Anca Chereches. 2014. A Prosodic Analysis of Romanian Pronominal Clitics. University of Pennsylvania Working Papers in Linguistics, volume 20. Available at https://repository.upenn.edu/pwpl/vol20/iss1/7/.

Noam Chomsky and Morris Halle. 1968. The Sound Pattern of English. Harper & Row.

Ann A. Copestake. 2001. Implementing typed feature structure grammars. CSLI lecture notes series.

Carmen Dobrovie-Sorin. 1999. Clitics across categories: The case of Romanian. In Henk C. van Riemsdijk, editor, Clitics in the Languages of Europe. Mouton de Gruyter.

Carmen Dobrovie-Sorin and Ion Giurgea, editors. 2013. A Reference Grammar of Romanian. Volume 1: The noun phrase. John Benjamins.

Gerlach and Janet Grijzenhout, Clitics in Phonology, Morpholo
John Benjamins.

Ciprian-Virgil Gerstenberger. 2
based generation pre-realization m
In Proceedings of EUROLA
2007. University of Iași, Romani
at https://www.researchgat
ion/233951937_A_morphology-b
realization_model_for_
tion.

Ciprian-Virgil Gerstenberger.
database and weak pronoun gene
Languages at the Crossroads:
creditation and Context of Use
of the 35th Edition of the Inte
ference of the Spanish Associa
Linguistics, pages 215–226. Universi
Available at https://www.rese
blication/332671041_A_Gramma
manian_weak_pronoun_generati

Ciprian-Virgil Gerstenberger.
are romanian clitic pronouns? Nordlyd,
47(1):37–57.

Bruce Hayes, Bruce Tesar, and Ki
OTSoft 2.5 [software package].
guistics.ucla.edu/people/hay

Ronald M. Kaplan, John T. Maxwel
Holloway King, and Richard Crou
tegrating finite-state technolo
grammars. In Proceedings of the ESSL
Workshop on Combining Shallow a
cessing for NLP.

Udo-Michael Klein. 2007. Encoding of argu
ment structure in Romanian an
Ph.D. thesis, University of Lo
at http://hdl.handle.net/1185
0012-8EEC-D.

Kimmo Koskenniemi. 1983. Two-le
morphological analysis. In IJCAI, volume 83,
pages 683–685.

Géraldine Legendre. 2001. Posit
verbal clitics at PF. In (Ger
zenhout, 2001).

Paola Monachesi. 2001. Clitic p
Romanian verbal complex. In (Ge
zenhout, 2001).

Paola Monachesi. 2005. The Verbal Complex
Romance: A Case Study in Gramma
faces. Oxford University Press.

John J Ohala. 2008. The Emergent s
syllable in speech. Taylor & Francis

OpenAI. 2024. ChatGPT [Large La
https://chat.openai.com

Alexandra Popescu. 2000. The morphophonology of the Romanian clitic sequence. In Lingua, volume 110, pages 773–799. Elsevier.

Alexandra Popescu. 2003. Morphophonologische Phänomene des Rumänischen. Ph.D. thesis, University of Düsseldorf. Available at https://docserv.uni-duesseldorf.de/servlets/DocumentServlet?id=3187.

Kan Sasaki and Daniela Căluianu. 2000. An Optimality Theoretic Account for the Distribution of Pronominal Clitics in Romanian. Technical report, University of Tsukuba. Available at https://www.academia.edu/33668351/An_optimality_theoretic_account_for_the_distribution_of_pronominal_clitics_in_Romanian.

Stanca Somesfalean. 2007. On the Form and Interpretation of Clitics. Ph.D. thesis, Université du Québec à Montréal. Available at https://archipel.uqam.ca/9628/.

Oana Săvescu Ciucivara. 2009. A Syntactic Analysis of Pronominal Clitic Clusters in Romance - The view from Romanian. Ph.D. thesis, New York University. Available at https://www.proquest.com/docview/304954033.