Enhancing a multi-faceted Breton verb-centered resource to help a language learner

Annie Foret
Univ. Rennes and IRISA, France
annie.foret@irisa.fr

Erwan Hupel
Univ. Rennes 2 and CELTIC-BLM, France
erwan.hupel@univ-rennes2.fr

Pêr Morvan
An Drouizig, France
per.morvan.bzh29@gmail.com

Abstract

This article builds on two recent resources for Breton, a verb-centered database and a set of sentences in the universal dependencies (UD) format. Our focus is on Breton, an endangered language in the Celtic family. We provide an analysis of the resource on verbs and show how it can be connected and transformed to a multi-faceted system intended to help a learner in a flexible way. We discuss several scenarios.

1 Introduction and objectives

Working on low-resourced languages comes with specific challenges (Vergez-Couret et al., 2024).

In this paper, we consider this issue for Breton; a discussion can be found for example in (Foret et al., 2015). We provide here a workflow that aims to facilitate the use and access by a learner to rich linguistic data, in a flexible way. The workflow is intended to be open, reproductible, with easily adaptable outputs[1]. The prototype is also devised for a use case within the *formal concept analysis* (FCA) paradigm[2], handling several facets (kinds of information).

In contrast to a *carrier sentences* or *seed sentence approach* (Heck and Meurers, 2022), our interface starting-point is a set or subset of verb infinitives that a learner wishes to master (by viewing information in several prepared hierarchies) or that he may simply discover by serendipity or *incidental learning* (Renduchintala et al., 2019). The outputs are thus intended as self-assisted learning systems involving small resources and lightweight technology (a browser, online or offline) but not requiring specific technical knowledge from users. We use a relational database system to store the linguistic data and generate the end systems proposed to a learner, but this is hidden to the users.

The plan of paper is as follows. In section 2 we point to some Breton specifities and difficulties; in section 3 we discuss two different kinds of resource available for Breton; these resources are used in the new workflow described in section 4; section 5 discusses scenarios enabled by the resources and workflow; section 6 concludes with perspectives.

2 Breton linguistic features

In Breton syntax, the verb occurs as second constituant and allows one to put the most important first. Consonant mutations are a particularity of Breton and other languages in the Celtic family. Depending of grammatical features and other features, some initial consonants change to others (Hupel, 2021; Jouitteau, 2009-2024). This is a difficulty for Breton learners and automatic processing as well.

The Breton verb varies depending on a lot of elements. There are two main categories, related to conjugations or not related. Firstly, as it is common, conjugations vary according to person, number, tense, mood, aspect and voice. Secondly, a consonant mutation may apply. For example the initial ”k”[k] will become ”c’h”[x], or ”g”[g]: the infinitive ”kanañ” (EN: to sing) occurs as ”gan” in this sentence ”An eostig a gan bemnoz” (EN: The nightingale sings every evening) where a *soft*

[1] available at https://gitlab.inria.fr/foret/termlis/-/tree/main/Breton/Verbs

[2] in this project: https://www.smartfca.org/

Annie Foret, Erwan Hupel and Pêr Morvan. Enhancing a multi-faceted verb-centered resource to help a language learner: the case of breton. *Proceedings of the 13th Workshop on Natural Language Processing for Computer Assisted Language Learning (NLP4CALL 2024)*. Linköping Electronic Conference Proceedings 211: 59–66.

mutation from ”k” to ”g” is induced by the preceding ”a” verbal particle; the other verbal particle ”e” yields a *mixed mutation*, as in ”Bemnoz e kan an eostig”. The ”pa” (EN: when) conjunction yields a *soft mutation*: the initial ”k”[k] will become ”g”[g], as ”kregiñ” (EN: to begin) in Figure 4. But conjunction ”ma” (EN: if) yields a *mixed mutation* where the verb initial ”k”[k] is unchanged.

Other difficulties may arise for: a verb without ending (”kemer”, ”lenn”, ”komz”); an altered verb base in the infinitive (”skeiñ” → sko-, ”mervel” → marv-); a verb base different from the infinitive (”gounit” → gounez-, ”dont” → deu-).

Most Breton verbs are regular[3] (Desbordes, 1999). Nevertheless, several grammatical categories are distinguished in the database see section 3 and Figure 2.

3 Two existing Breton resources

Breton is a low-resourced language. Nevertheless, we discuss two resources of valuable interest in this verb-centered proposal.

3.1 The DVB verb database

The DVB Breton verb site is handled by the association An Drouizig. Figure 1 shows the top of the page for the Breton verb ”kanañ” (EN: to sing) [4]. Each page (in BR, FR, or EN language) provides different kinds of informations on a given verb:

- tags (such as ”European level A1 verb”), other forms, translations, sources, links are on the top,
- mutation modes and examples,
- conjugations in details.

We give below a simplified explanation of the DVB verb group classification (for conjugation):
- most verbs are regular and in $d1$;
- verbs in $d2$ are regular, end in -aat/-at and express an action taking place;
- verbs in $d3$ are regular, end in -a and express picking up something etc.;
- verbs in $d4$ are semi-regular, end in -iañ, -iiñ;
- verbs in $d5$ are semi-regular, end in -liañ, -liiñ;
- verbs in $d6$ are semi-regular, end in -niañ, -niiñ;
- verbs in $d7$ are irregular, follow the conjugation

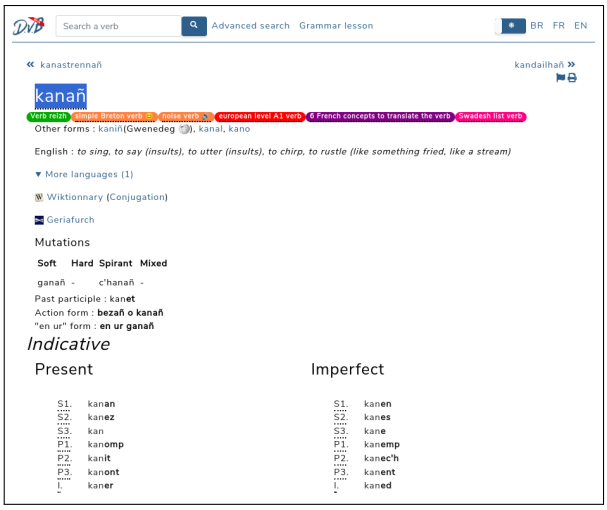

Figure 1: ”kanañ” (to sing) at https://displeger.bzh

of ”ober”;
- $d8$ regroups special verbs: ”bezañ” / ”bout” (to be), ”kaout” / ”endevout” (to have), ”dont” (to come), ”mont” (to go), ”gouzout” (to know).

Figure 2 shows statistics on grammatical categories. We computed them on the DVB relational database for Breton verbs provided by An Drouizig, behind the DVB website.

category	nb_infinitive	nb_verb_id	example
bezan	2	1	bezañ
d1	11294	9308	abafañ
d2	838	823	abafaat
d3	487	471	abona
d4	562	542	adeouliañ
d5	101	99	adiziliañ
d6	100	98	adkoaniañ
d7	13	12	addizober
dont	6	3	addonet
gouzout	2	1	goût
kaout	2	1	endevout
mont	3	2	enmont

Figure 2: Verb grammatical categories in DVB.

More globally, our analysis of the relational database yields the general data model in Figure 3 with content statistics. Each table comes with a name, a list of attributes (the table columns) with its number of lines at the top. The underlined attribute is its *primary key*. Each edge stands for a *foreign key* connecting an attribute in the source table to another attribute that it refers to in the target table.

This database is a key component for the workflow proposed in section 4. We now describe the second component used in the workflow.

[3] see also https://arbres.iker.cnrs.fr/index.php?title=Verbes_irr%C3%A9guliers

[4] https://displeger.bzh/en/verb/kana%C3%B1

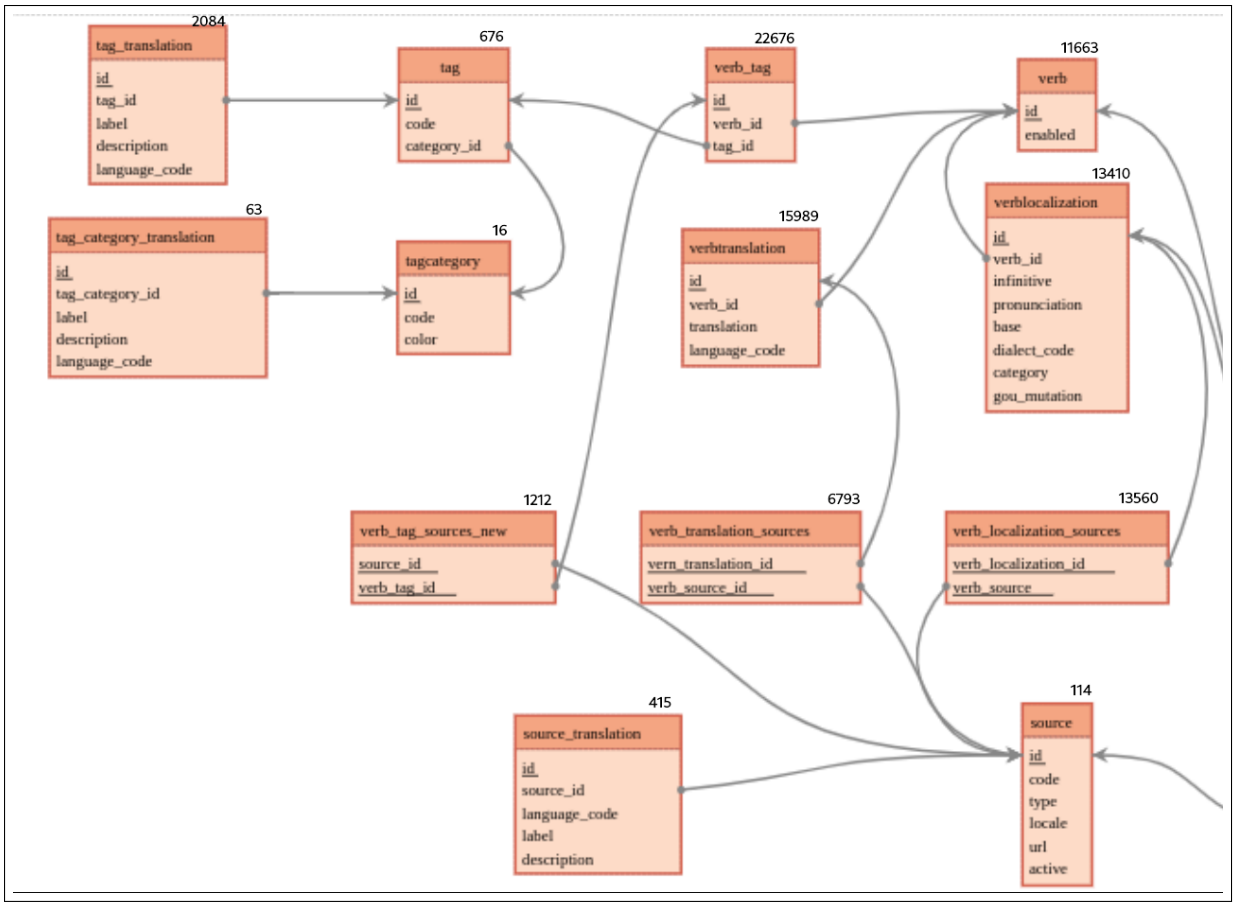

Figure 3: A relational schema for DVB (main part): each table has a name, a line count above, attributes (the key is underlined). The edges show foreign keys. We focus particularly on `tag`, `verbtranslation` and `verblocalization` tables and their `code`, `translation`, `infinitive`, `category` attributes.

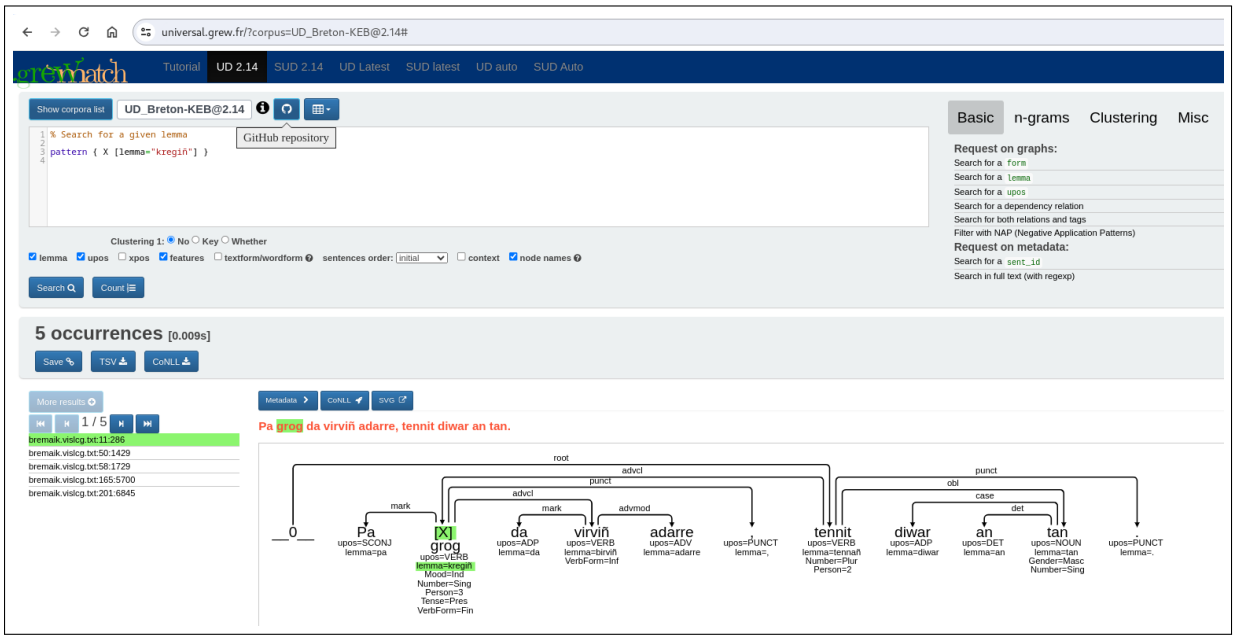

Figure 4: `Grew` can show (and rewrite) the treebank part that matches a linguistic pattern (on the top). The infinitive ”kregiñ” (EN: to begin) is highlighted in this parse tree (its exact form is ”grog”, with soft mutation after ”Pa”). The text[eng] metadata is: ”When boiling again, draw from the fire”. The second infinitive is ”birviñ” (EN: to boil).

3.2 UD treebanks

Dependency syntax has been developped for a long time, for example in (Mel'čuk, 1988). This approach underlies the active area of universal dependencies (Nivre et al., 2016; de Marneffe et al., 2021), an annotation framework with treebanks in over 150 languages.

Several dependency treebanks are developed for Celtic languages (Lynn and Foster, 2016; Batchelor, 2019; Heinecke and Tyers, 2019). In this work we consider the UD_Breton-KEB corpus V1.0[5] (Tyers and Ravishankar, 2018) with a 2023 revised version [6]. Its first annotated sentence is:

```
# sent_id = apertium.vislcg.txt:1:0
# text = N'int ket aet war-raok.
# text[eng] = They didn't progress.
# text[fra] = Ils n'ont pas progressé.
# labels = to_check
1   N'  ne  ADV adv Polarity=Neg    4   advmod  _   SpaceAfter=No
2   int bezañ   AUX vblex
Mood=Ind|Number=Plur|Person=3|Tense=Pres|VerbForm=Fin   4   aux _   _
3   ket ket ADV adv _   4   advmod  _   _
4   aet mont    VERB    vblex   Tense=Past|VerbForm=Part    0
root    _   _
5   war-raok    war-raok    ADV adv _   4   advmod  _   SpaceAfter=No
6   .   .   PUNCT   sent    _   4   punct   _   _
```

We see the meta-information above at the beginning (lines starting with #), then a line by word occurrence in sentence order, with tabs separated columns: ID FORM LEMMA UPOS XPOS FEATS HEAD DEPREL DEPS MISC. From this, the dependency parse tree can be drawn (as in Figure4). Our workflow exploits meta-information and FORM, LEMMA, UPOS columns (crucially, UPOS='VERB' tells which word occurrences are verbs).

The universal dependencies site also collects a list of tools for working with UD. We mention two:
- Grew[7] (Guillaume, 2021) is a graph rewriting tool dedicated to applications in Natural Language Processing. Figure 4 shows the Breton treebank with a query.
- CoNLL-U viewer at rug.nl is a simple browser-based UD viewer. Figure 9 displays a rewritten version of the treebank and its browsing.

We will show a scenario linking verb data to related sentences that have been parsed in the universal dependency format. The workflow enabling this scenario is described in next section.

[5] available at https://universaldependencies.org/treebanks/br_keb/index.html

[6] at https://github.com/UniversalDependencies/UD_Breton-KEB

[7] https://grew.fr

4 A new workflow

The database is analyzed and processed to control, to enhance and to select appropriate fragments. The workflow outputs several versions (.csv, .html, .ttl/rdf exports) allowing different scenarios.

DVB processing We define different views on the DVB tables. For an HTML output, a typical generated line (in the tag part) is:

```
<a href="https://displeger.bzh/fr/verb/selaou"
data-id="5711" title="écouter" data-init="s"
data-categ="d1" data-base="selaou" data-idV="38435"
data-tag="verb_al_live_A1" data-row="179">selaou</a>
<a class="bis" href="#selaou-1" > # </a>
```

Some informations are rendered as HTML attribute-value pairs, for a basic HTML view with many CSS stylesheet possibilities, this is also shaped to show useful information on hover and useful links (to a relevant DVB page or to relevant UD sentences). The grammatical category stored in verblocalization is given as an attribute-value pair, such as data-categ="d1" for the verb "selaou", not visible in the browser, but could appear as HTML content, by a simple CSS rule.

Verb facet selection The DVB database contains grammatical categories in the verlocalization table. DVB also contains many tags of various kinds in the tag table, that we organize in 11 subclasses (such as level, or domain)[8] to view fragments in a flexible and informative way.

We produce in this way an enhanced version (see Figure 5) in turtle/RDF (semantic web) format, that enables search using the SPARQL Query Language for RDF or related tools such as Sparklis (Ferré, 2016); this approach is applied to Georgian verbs by (Ducassé and Elizbarashvili, 2022).

For ease of use, we also produce simpler HTML versions: either for verbs alone as in Figure 6, or for verbs connected with sentences from a UD treebank (parse trees) as explained below.

Treebank preparation Before an upload in SQL, the UD corpus file is prepared, adding a column with a line number, and handling special symbols (quote

[8] our current tag subclass list is: Level, Domain, Link, Substring, NbSyllabs, Ends (for verb ending), Change (for variations), Construct, Args (for transitive, etc.), Synset (number of synonyms for translation), Other.

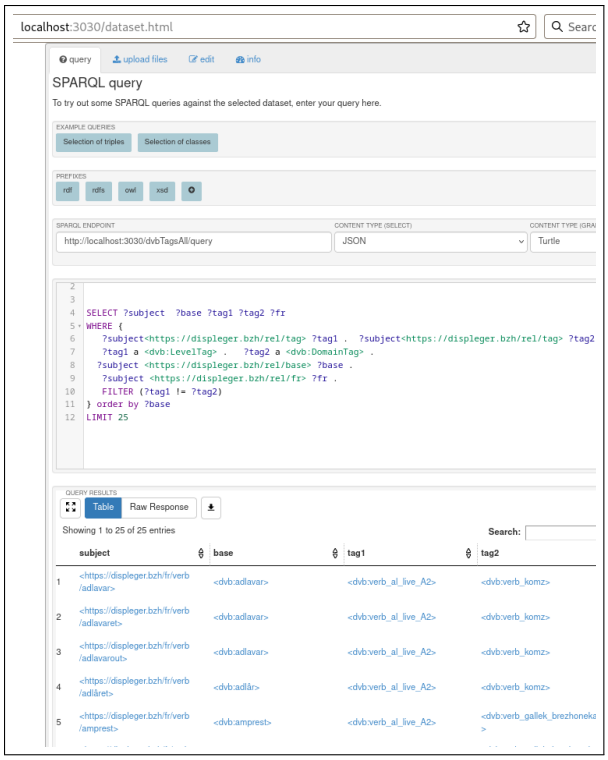

Figure 5: DVB verb facets in RDF, with tags hierarchy.

in quote, etc.). We then partition the numbered lines in two relational tables, `br_keb_ud(line,WID,FORM,LEMMA,UPOS, XPOS,FEATS,HEAD,DEPREL,...)` for word information, `br_keb_sent(line, sent)` for whole sentence information. We then build SQL views that generate HTML lines such as:

```
<p class="sent"><a id="selaou-2"
href="https://displeger.bzh/fr/verb/selaou">selaou</a>
<span data-form="selaouit" data-root="[r]">selaouit
</span> <span class="line" data-sentnum="511">
(line 8665, sent 511) </span># text = Va
<span class="solution" title="selaouit">_</span>
<span class="w" title="selaouit">selaouit</span>!
<span class="sentFr"> # text[fra] = Écoutez-moi!
</span><p>
```

We use in particular this CSS rule:

```
span.w {visibility: hidden;}
```

to hide the conjugated form of the verb (this CSS rule may be dropped to show full sentences).

A similar output is generated for English translations, selecting lines of `br_keb_sent` containing "text[eng]" instead of "text[fra]".

5 Enabled use case scenarios

We first describe successive scenarios in the HTML mode, based on a navigator. We suppose

Figure 6: DVB verbs in HTML, <details> tag "european level A1" open. The top tag contents are hidden. After a click on the "verb al live A1" tag, the related verb list is visible. A translation is then directly visible by hovering over a verb.

the user has a local copy of the HTML and CSS files. In this HTML mode, the file produced by the workflow gathers several kinds of information in one place. And the user can view and interact without internet connection.

Example scenario (1): verbs only, HTML mode The first file version regroups information from the DVB database, showing verbs by categories (such as "level A1", then by initial letter inside a category). As explained in section 4, each verb is accompanied with attribute-value pairs that can be shown or not, depending on the chosen CSS rules (on the developer side). Figure 6 corresponds to this version. This rendering uses the HTML5 <details> element, so that the user can open and close an item (such as the "verb al live A1" tag) to display its content. For each infinitive such as achivañ: its translation appears on hover and a click on the infinitive links the DVB site for this verb.

An enhanced version is provided that regroups the various tags in 11 classes (supertags) as explained in section 4 and visible in the middle of Figure 7. Note that this is strict hierarchy on tags, while verbs appear in each tag they belong to (they may appear several times in the file).

In these ways, users may choose a tag or a facet (possibly several) and test their knowledge on verbs belonging to this tag or this facet, in a compact way.

Example scenario (2): including sentences (HTML mode) This second file has two sections, a tag section corresponding to the whole first

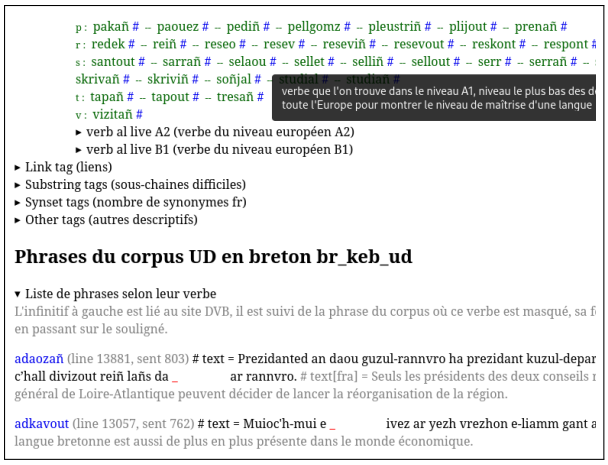

Figure 7: DVB verb facets in HTML, <details> tag "european level A1" open, with sentences. In the sentence section, the infinitive on the left is linked from the A1 tag list and links to the DVB site.

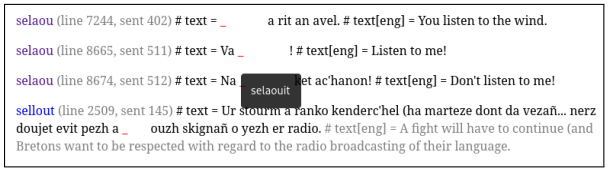

Figure 8: Breton sentences from the corpus with their EN translation and the verb form hidden (shown on hover), linked from the selected infinitive "selaou" in the DVB tag "european level A1" open.

file, and a sentence section as in Figure 7. Sentences in the sentence section are ordered by infinitive (rewritten on the left of the sentence). A click on # in the tag section points to the first sentence (in the sentence section) where the verb occurs; the verb occurrence is hidden (by the chosen CSS stylesheet) in the sentence. The exact verb form appears on hover. Sentences with a same infinitive follow each other, which enables training on the same verb with proximate sentences.

Note that a sentence appears for each infinitive that occurs in it (a sentence may appear several times in the file, depending on its number of verbs).

In these ways, users may choose a tag or a facet (possibly several) and practice or check several aspects on a verb:
- its hidden meaning as in the first file;
- its hidden conjugation in a set of sentences (the solution appears on hover over the key _, as "selaouit" in Figure 8). For explanations, they may also consult the appropriate page of the DVB site, by a click on the infinitive (in either section).

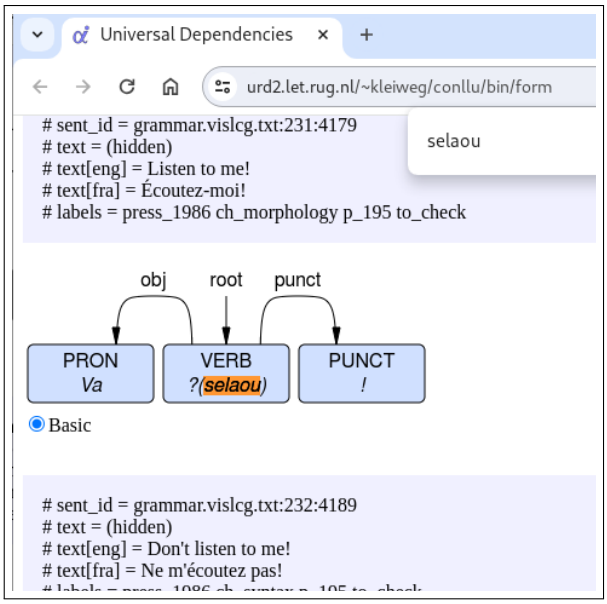

Figure 9: Browsing the rewritten UD treebank (the infinitive in the tree node replaces the exact verb form).

Example scenario (3): including parse trees (HTML mode) Grew can show and rewrite a treebank part that matches a linguistic pattern.

On the preparation side, to transform the treebank we applied the Grew command[9] to rewrite tree features and then converted the output with the sed command to hide the "text =" metadata. The following one-rule rewriting system hides all verb-forms, replaced by their infinitive :

```
package hide-verb-form {
 rule hideVerbsForm {
   pattern { X1 [upos=VERB] }
    without {X1.mark = "x"}
   commands {X1.mark = "x" ;
             X1.form = "?(" + X1.lemma + ")"; }
  }
}
strat main { Onf(hide-verb-form) }
```

On the user side, the resulting treebank can then be loaded and searched as in Figure 9 for a small sentence (see Figure 4 for a larger tree).

Semantic Web mode In this mode, we assume the user has a copy of the .ttl file and has installed a SPARQL server such as Apache Jena Fuseki. The user or developper familiar with RDF web semantic standards can load the .ttl file to query it and to explore the data in these two ways: directly write a SPARQL query (as exampli-fied in Figure 5 or build a query with Sparklis that is a tool with guidance in a natural language as in (Ferré, 2016; Ducassé and Elizbarashvili,

[9] a rewriting system may also be loaded on web.grew.fr and applied on the selected corpus tree

2022). In this mode also, the file produced by the workflow gathers several kinds of information in one place. The facet filtering is very flexible, even more with `Sparklis` requiring less knowledge on the data model.

6 Conclusion and future work

The workflow described in this paper outputs easy-to-use language learning verb-centered contexts, aimed to help a Breton learner. The outputs gather heterogeneous information on one or few files, so that a user may train with different scenarios and facets of verbs (including flat or structured sentences). This is still work in progress, more automation and scenario variations could be provided and tested. A user study could also be added. We list some other points for future work.

- At the level of word descriptors, the hierarchy of tags could be exploited in a more elaborated way, in particular within the *formal concept analysis* (FCA) paradigm; we could test the potential of such approaches on the design and use of a self-assisted learning system. The FCA approach could also show sets of verbs sharing a same set of descriptors. Sentences from a parsed corpus (where verbs are tagged with their linguistic features as in `UD`) could inherit their verb descriptors as well, providing indicators per sentence in a flexible way.

- At the level of treebanks, a new[10] Breton `UD` treebank is in preparation, which may provide new insights. Other sentence structures have been proposed depending on the preferred grammatical formalism and parsing principles; the `SUD` (Surface-syntactic Universal Dependencies) variant (Gerdes et al., 2018) is available for `UD` treebanks from `Grew` and could have been proposed here instead of `UD`. Semantic structures such as `AMR` (Abstract Meaning Representations)[11] (Heinecke and Shimorina, 2022) might also bring help, but we are not aware of such data for Breton.

- As concerns workflow handling, the development follows a reproducibility principle and we believe the workflow should apply to the new treebank and to augmented versions of the verb database (with few adjustments).

We generated browser-based versions aimed at individualized learning solutions. Worksheets or gap filling exercises could be generated in a close way by the workflow.

We think there is a need to enhance existing resources especially on a low-resourced and endangered language such as Breton. We hope this development is a step in this direction.

Acknowledgments.

We thank Gwenn Meynier from An Drouizig, for information on the DVB resource for Breton.

Annex: Breton mutation system

See Figure 10 for an overview on the four mutation kinds: soft, spirant, hard and mixed.

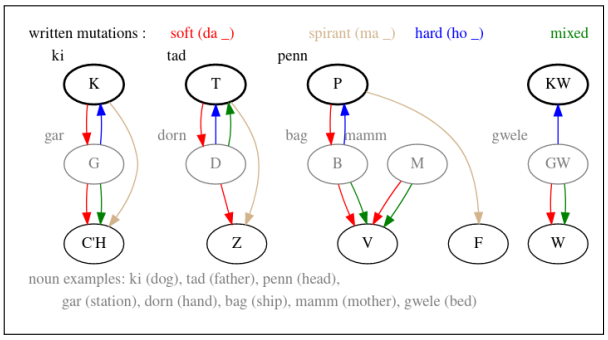

Figure 10: Breton initial mutation overview (on nouns)

References

Colin Batchelor. 2019. Universal dependencies for Scottish Gaelic: syntax. In *Proceedings of the Celtic Language Technology Workshop*, pages 7–15, Dublin, Ireland.

Yann Desbordes. 1999. *Petite grammaire du breton moderne*. Mouladurioù Hor Yezh.

Mireille Ducassé and Archil Elizbarashvili. 2022. Finding lemmas in agglutinative and inflectional language dictionaries with logical information systems: The case of Georgian verbs. In *Proceedings of XX EURALEX International Congress*. Mannheim: IDS-Verlag. Demonstration.

Sébastien Ferré. 2016. Sparklis: An expressive query builder for sparql endpoints with guidance in natural language. *Semantic Web*, 8:405–418.

[10] see `https://arbres.iker.cnrs.fr/index.php?title=Breton_treebank_II`

[11] `AMR` page : `https://amr.isi.edu/`, `AMR` bibliography `https://nert-nlp.github.io/AMR-Bibliography/`

Annie Foret, Valérie Bellynck, and Christian Boitet. 2015. Akenou-breizh, un projet de plate-forme valorisant des ressources et outils informatiques et linguistiques pour le breton. In *Actes de la Traitement Automatique des Langues Régionales de France et d'Europe*, Caen, France. Association pour le Traitement Automatique des Langues.

Kim Gerdes, Bruno Guillaume, Sylvain Kahane, and Guy Perrier. 2018. SUD or surface-syntactic Universal Dependencies: An annotation scheme near-isomorphic to UD. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 66–74, Brussels, Belgium. Association for Computational Linguistics.

Bruno Guillaume. 2021. Graph Matching and Graph Rewriting: GREW tools for corpus exploration, maintenance and conversion. In *EACL 2021 - 16th conference of the European Chapter of the Association for Computational Linguistics*, Kiev/Online.

Tanja Heck and Detmar Meurers. 2022. Generating and authoring high-variability exercises from authentic texts. In *Proceedings of the 11th Workshop on NLP for Computer Assisted Language Learning*, pages 61–71, Louvain-la-Neuve, Belgium. LiU Electronic Press.

Johannes Heinecke and Anastasia Shimorina. 2022. Multilingual Abstract Meaning Representation for Celtic languages. In *Proceedings of the 4th Celtic Language Technology Workshop within LREC2022*, pages 1–6, Marseille, France. European Language Resources Association.

Johannes Heinecke and Francis M. Tyers. 2019. Development of a Universal Dependencies treebank for Welsh. In *Proceedings of the Celtic Language Technology Workshop*, pages 21–31, Dublin, Ireland. European Association for Machine Translation.

Erwan Hupel. 2021. Le Breton [quelques contrastes pertinents entre le français et le breton]. *In projet Langues et grammaires du monde dans l'espace francophone*. `lgidf.cnrs.fr`.

Mélanie Jouitteau. 2009-2024. ARBRES, wikigrammaire des dialectes du breton et centre de ressources pour son étude linguistique formelle, IKER, CNRS. `http://arbres.iker.cnrs.fr`. Licence Creative Commons BY-NC-SA.

Teresa Lynn and Jennifer Foster. 2016. Universal dependencies for irish. In *Proceedings of the Celtic Language Technology Workshop*, Paris, France.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

I. Mel'čuk. 1988. *Dependency Syntax*. SUNY Press, Albany, NY.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).

Adithya Renduchintala, Philipp Koehn, and Jason Eisner. 2019. Simple construction of mixed-language texts for vocabulary learning. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 369–379, Florence, Italy. Association for Computational Linguistics.

Francis M. Tyers and Vinit Ravishankar. 2018. A prototype dependency treebank for breton. In *Actes de la Conférence TALN. CORIA-TALN-RJC 2018 - Volume 1 - Articles longs, articles courts de TALN, Rennes, France, May 14-18, 2018*, pages 197–204. ATALA.

Marianne Vergez-Couret, Delphine Bernhard, Michael Nauge, Myriam Bras, Pablo Ruiz Fabo, and Carole Werner. 2024. Managing fine-grained metadata for text bases in extremely low resource languages: The cases of two regional languages of France. In *Proceedings of the 3rd Annual Meeting of the Special Interest Group on Under-resourced Languages @ LREC-COLING 2024*, pages 212–221, Torino, Italia. ELRA and ICCL.