Introducing the Research Constructicon as a knowledge graph database for Construction Grammar

Elodie Winckel
Friedrich-Alexander-Universität Erlangen-Nürnberg
elodie.winckel@fau.de

Peter Uhrig
Friedrich-Alexander-Universität Erlangen-Nürnberg
peter.uhrig@fau.de

Stephanie Evert
Friedrich-Alexander-Universität Erlangen-Nürnberg
stephanie.evert@fau.de

Abstract

This paper introduces the Research Constructicon, a constructicographic tool designed for linguists working in Construction Grammar. It documents research from a Research Training Group in the form of a network of construction (in the sense of Construction Grammar). The Research Constructicon is born from a fruitful collaboration across subfields (e.g., cognitive linguistics, diachronic linguistics, neurolinguistics). Unlike typical reference constructicons, the Research Constructicon leverages Semantic Web technologies. Its architecture combines an ontology, an RDF database, and a web interface, where contributors can document constructions without expertise in RDF. The paper outlines the theoretical alignment with Construction Grammar, the technical implementation, and the web interface's features.

1 Introduction

This paper presents the Research Constructicon (RCxn), a tool designed by linguists for linguists, with a specific focus on Construction Grammar. Construction Grammar posits that linguistic knowledge is organized in constructions, i.e. more or less schematic chunks that pair a formal side (e.g., phonology or syntactic patterns) with a functional side (encompassing semantic and pragmatic knowledge). These constructions are mostly modeled as a network, referred to as a constructicon. Cognitive strands in Construction Grammar hypothesize mental constructicons as a cognitive reality. On the model of mental constructicons, some teams have created reference constructicons, i.e. collections of construction descriptions which aim to systematically capture linguistic knowledge across languages (Lyngfelt et al. 2018a; for a discussion of the distinction between mental and reference constructicons, see Herbst 2019).

Our project contributes to this field but diverges from previous efforts. The context for its development is a Research Training Group (RTG) at the Friedrich-Alexander-Universität Erlangen-Nürnberg, where professors and early-career researchers from various linguistic departments—including German, English, Romance, and Arabic—work together under the common framework of Construction Grammar. Over the past four years, this RTG has brought together nearly 30 doctoral candidates, some of whom have already completed their degrees. The need to consolidate the diverse research outputs of these projects into a cohesive, accessible resource led to the creation of the RCxn. At its core, the RCxn is a database that aggregates data from all doctoral projects, providing a centralized platform.

In this paper, we outline the theoretical and technical foundations of the RCxn, emphasizing how its design aligns with the principles of Construction Grammar while leveraging Semantic Web technologies to ensure flexibility and interoperability (Section 2). We then describe the architecture of the database (Section 3) and the web interface that facilitates access and contribution to the database (Section 4).

2 Theoretical and Technical Foundations

Reference constructicons are typically designed as general descriptive resources or as pedagogical resources for foreign language learners and teachers (Lyngfelt, 2018; Herbst, 2019). In some cases, their computational usability is also highlighted as a key motivation. In contrast, the RCxn is conceived as a repository of actual linguistic research output, reflecting the empirical and theoretical findings in the field.

2.1 The Case for Semantic Web technologies in constructicography

Several notable reference constructicons have been developed before. These include the *English*

Constructicon at the University of Birmingham (https://englishconstructicon.bham.ac.uk; Perek and Patten 2019), the *CASA-ConstructiCon of the English Language at Friedrich-Alexander Universität Erlangen-Nürnberg* (https://constructicon.de/; Herbst et al. 2023–; Herbst and Hoffmann 2024), the FrameNet-Konstruktikon des Deutschen for German at Heinrich Heine University Düsseldorf (https://framenet-constructicon.hhu.de; FKD; Ziem and Flick 2019), the FrameNet Brasil Constructicon for Brazilian Portuguese at the Federal University of Juiz de Fora (https://webtool.frame.net.br/report/cxn; Torrent et al. 2018), the Svenskt konstruktikon for Swedish at the University of Gothenburg (https://spraakbanken.gu.se/karp/; Lyngfelt et al. 2018b), the Russian Constructicon at the Arctic University of Norway (https://constructicon.ruscorpora.ru/; Bast et al. 2021), the Ukrainian Constructicon at the Arctic University of Norway (https://constructicon.github.io/ukrainian/; Palii et al. 2023) and a planned Norwegian Constructicon (Endresen and Mikkelsen, 2024). Additionally, there are related projects focused on building a FrameNet for diverse languages.

These reference constructicons have historically used relational databases, which offer high performance but are inherently rigid due to their tabular structure. In this respect, our current project diverges significantly by adopting Semantic Web technologies, which supports more flexible descriptions. Furthermore, ontologies are a core element in this framework, which in turn allows our database to be used for automatic reasoning. But this framework offers more than just technical advantages; it is also of great theoretical interest in that it allows us to come as close as possible to the way researchers conceptualize the organization of linguistic knowledge.

2.1.1 Knowledge graphs can represent linguistic knowledge best

Construction Grammar views linguistic knowledge as forming a constructicon, i.e. a network. There is a wide range of research dealing with the organization of this network. Our project is inspired by Diessel's model, who proposes that linguistic constructions display network structures both internally and externally (Diessel, 2019, 2023). Internally, each construction forms a network where elements are interconnected through functional and semantic features as well as sequential dependencies. Externally, constructions are embedded within a broader network and function as nodes linked by derivational processes or shared similarities such as phonetic characteristics.

Knowledge graphs offer a structured yet flexible framework that mirrors Diessel's conceptualization more accurately than table-based relational databases. Knowledge graphs were conceived at the crossroads of semantics and mathematics to store description of world knowledge in a structured way, as directed graphs. Nodes in such a graph are instances of classes that are defined in a vocabulary. Edges in the graph are properties that are also defined in the vocabulary. They enable the modeling of both the internal structure of constructions and the relationships between different constructions, providing a theoretically grounded representation.

2.1.2 Diversity in linguistic research requires flexibility

Semantic Web technologies have long been recognized in the digital humanities as a powerful tool for addressing data heterogeneity. For example, Langmead et al. (2016) argued that this framework enabled the comparison and aggregation of complex scientific data without imposing rigid, universal standards. Their goal was to link disparate historical projects together.

Linguistics research is also very fragmented and diverse. The doctoral candidates in our RTG work on different languages and with different approaches. Although they all work in Construction Grammar, their research covers a wide range of subfields in linguistics, including contrastive linguistics, cognitive linguistics, language pedagogy, diachronic linguistics, computational linguistics, and neurolinguistics. We also want our tool to be usable for future projects whose nature we cannot yet foresee. Since the academic staff at universities is constantly changing, we do not even know which colleagues will join our own project in the years to come.

Even within constructicography, approaches vary widely. A quick glance at the Constructicons mentioned above makes it quite obvious. Semantic Web technologies are highly flexible because they impose minimal constraints. At its core, a knowledge graph consists of triplets, each forming an assertion. These triplets can be thought of as

cells in a relational database, where each cell is defined by a subject (row), predicate (column), and object (value). A core element of this framework is the use of ontologies (also called vocabularies), which ensure that every component of the triplets is clearly defined. Moreover, the ontology can be expanded with new resources (classes or properties) to meet evolving needs. The relationships between these new concepts and existing ones can also be documented within the ontology itself. This makes the framework ideal for projects that evolve organically.

2.1.3 The Semantic Web framework ensures FAIR Principles

The Semantic Web framework is inherently aligned with the FAIR principles (Findable, Accessible, Interoperable, and Reusable; Wilkinson et al. 2016).

- Findable: The Semantic Web uses Internationalized Resource Identifiers (IRIs) to assign persistent, globally unique identifiers to resources. They can therefore be easily referenced. These resources are linked to each other based on RDF (Resource Description Framework) standards, which represents data as triplets (subject-predicate-object). This structured format allows machines to parse, interpret, and reason over data. Not only the assertional knowledge (e.g., in our case the description of constructions) but the terminological knowledge is richly documented in ontologies, and follows the RDF standards. These ontologies provide formal definitions of terms, enabling machines to understand relationships and infer new knowledge.

- Accessible: The Semantic Web provides an architecture for publishing structured data on the web using open licenses, in RDF format, with IRIs as identifiers, and linked to other datasets, a practice commonly referred to as Linked Open Data. By doing so, we ensure that both humans and machines can retrieve data. The RCxn dataset is published under the Creative Commons Attribution 4.0 International (CC-BY 4.0) license. Additionally, its data is hosted on a publicly accessible repository (`https://github.com/ElodieWinckel/RCxn`). We use exclusively open and standardized RDF serialization formats (RDF/Turtle and RDF/XML).

- Interoperable: The Semantic Web framework promotes the use of shared ontologies for better interoperability. The RCxn dataset leverages already existing ontologies, including RDF and RDF Schema (RDFS) for foundational RDF schema definitions, Dublin Core (DB) for general metadata, FOAF (Friend of a Friend) to describe researchers contributing to the project and their research questions, OLiA (Ontologies of Linguistic Annotation; Chiarcos and Sukhareva 2015) for the description of linguistic features (e.g., semantic roles or tense), and the OntoLex module for FRequency, Attestations and Corpus data (FrAC; Chiarcos et al. 2022) for the description of corpora and frequencies. Since the data is machine-retrievable, data can be aggregated from multiple datasets.

- Reusable: Semantic Web technologies encourage the use of open standards and linked data principles to ensure that data can be combined, enriched, or repurposed across domains without loss of context or meaning. By adhering to these principles (via the CC-BY license, use of already documented ontologies and a precise documentation of its own ontology), the RCxn ensures (and encourages) reuse of the data produced by the project.

2.2 Technical background for the Research Constructicon

The RCxn project has resulted in the creation of three components, which are closely intertwined.

The first of these is an **ontology**, which defines the conceptual schema: it specifies the classes, properties, and their logical relationships. This includes formal definitions of what each class and property represents, as well as domain and range restrictions, where the domain specifies the class to which a property applies, and the range defines the class of values the property can take. The ontology defines terms such as “Construction”, “Construction Element”, or “Metaphorical Link”. We have structured this ontology into several modules, each addressing specific aspects. The Research module, for example, defines terminology related to linguistic research (research questions, findings, etc.). The ontology modules were written in OWL using the Protégé software (Musen, 2015). This component is not the focus of this paper, which is primarily centered on the

database and its web interface. However, the reader is referred to Winckel (2025) and Winckel et al. (2026) for a detailed discussion. The full technical documentation of the ontology is available at https://github.com/ElodieWinckel/RCxn/blob/master/ontologies/ontology_doc.md. We will refer to the respective namespaces of these different modules throughout the paper in a manner common for RDF standards:

- the prefix `rcxn:` refers to the constructicography module where resources useful to describe constructions are defined
- the prefix `rsrch:` refers to the research module where resources useful to describe linguistic research are defined
- the prefix `links:` refers to the links module where resources useful to describe connections between constructions are defined
- the prefix `evid:` refers to the evidence module where resources useful to describe empirical evidence such as frequencies are defined

Note that this is not an exhaustive list of the modules. Our ontology contains further modules, like for example the comparative concept module which is an RDF implementation of the database of Comparative Concepts (https://comparative-concepts.github.io/cc-database/cc-database.html) proposed by Lorenzi et al. (2024; based on Croft, 2022) as a framework to align constructicons and so ensure comparability across constructicon projects. Some constructions in the RCxn make use of them.

The second entity created by this project is the **database** itself, a database of linguistic constructions and the research around them. The definition of classes and properties, which is handled by the ontology, has obviously a big impact on its architecture. It will be described in Section 3. The database is coded in RDF Turtle, an RDF format that, unlike XML/RDF, is much easier for humans to read. This format was chosen because it facilitates database manipulation when editing existing constructions. The implementation is in Python. We use the Python library `RDFLib` (Boettiger, 2018) for parsing and serializing RDF data.

The final component developed as part of this project is the **web interface**, which offers an intuitive and user-friendly way of interacting with the database content. This application was built using the widely-used Flask web framework in Python (Grinberg, 2018), which makes it easy for researchers to collaborate on the project. Furthermore, as Flask does not rely on any predefined database format, we are able to use the RDF database directly. We describe the web interface in Section 4.

3 The Database architecture

We call the part of the database dedicated to describing constructions the A-box, as opposed to the part of the database that contains the ontology. This name follows the terminology in Description Logic, which can be seen as setting up the theoretical foundations for Semantic Web technologies (although the latter goes beyond it, see Shi et al. 2005). Description Logic distinguishes between a T-box (Terminological Box) and an A-box (Assertion Box). While the ontology builds the T-box and is organized into different modules (see above), the A-box is stored in three separate files. One file (cx.ttl; prefix `cx:`) stores information about the constructions, one (membr.ttl; prefix `membr:`) stores information about the researchers and their research, and the last one (references.ttl; prefix `ref:`) contains the bibliography.

3.1 Description of a construction

At its bare minimum, a construction entry is a resource identified as an instance of the class `rcxn:Construction`. It must have a title and be linked to its language (see Section 3.4) and metadata. The metadata is itself a resource identified as an instance of the class `rcxn:Metadata`. This metadata resource must include, at minimum, a creation date and a link to the person who created the entry. Each of these assertions is stored as a subject-predicate-object triplet, meaning that a construction entry consists of at least six triplets (see Table 1).

Construction entries can be more complex. In Construction Grammar, constructions are defined as form-meaning pairings. Accordingly, in our database, the construction is linked to a functional-semantic component for the meaning side and to a sequence of one or more elements with a formal component for the formal side.[1]

[1]Note that this statement applies to verbal constructions, i.e., the majority of the constructions in the RCxn. It also contains, however, gestural constructions, whose description works differently. We do not cover gestural constructions in

Subject	Predicate	Object
X (resource)	is a (property)	Construction (class)
X (resource)	Title (property)	Title (literal)
X (resource)	Metadata (property)	Y (resource)
Y (resource)	is a (property)	Metadata (class)
Y (resource)	Date Created (property)	Date (literal)
Y (resource)	Annotator (property)	Z (resource)

Table 1: The six triplets required for a construction entry. The type of each value is indicated in parentheses. X is the URI associated with the construction described, Y the URI associated with the Metadata of this construction and Z the URI associated to a researcher.

The meaning(s) of a constructions are gathered under a resource belonging to the class `rcxn:ConstructionMeaning`. Most of our contributors chose to express the meaning as a literal text object, but it can also be represented using more structured data. Currently, the only implemented option for structured data is the addition of image-schemata.

A construction can consist of one or more elements, which might themselves be constructions: a word can consist of different morphemes, a phrase of different words and a clause of different phrases and words. Construction elements can be (lexically) specified or unspecified. Consider as an example the famous construction "The *X-er the Y-er*" (Fillmore, 1989; Hoffmann, 2019), in which *X* and *Y* stand for unspecified elements that may be filled by any linguistic material satisfying specific requirements, while *the* is fully specified. Construction elements can also be obligatory or optional. For example, the construction "*come + [noise/manner of movement] V-ing*" can be illustrated by sentences like (1).

(1) a. *Six-year-old Sally came running excitedly into the room.*

b. *People came running with lanterns and torches.*

The PP *into the room* in (1a) denotes the goal of the event. Its absence in (1b) indicates that it is optional, though it is a very common component of the construction (Patel, 2026). Some researchers might prefer to analyze both variants (with and without the optional element) as two different constructions but we allow our contributors to use optional elements if they wish. This means that optional elements with no change in meaning are also possible. An example in our RCxn is the construction '*X CONSIDER Y (as) Z*" in which the element *as* is optional, its absence not resulting in a change in meaning (Stampfer, 2024). This is controversial in the field. Some strands of Construction Grammar reject synonymy, positing that a change in form results in a change in meaning and/or function (e.g., Goldberg 1995; see Cappelle 2006 and Uhrig 2015 for counterarguments). Once again, we have decided to give our contributors the freedom to analyze constructions as they see fit.

The formal component of a construction element (an instance of the class `rcxn:SlotForm`) depends on the type of construction element. If the element is lexically specified, its formal component can describe its phonology or its surface form. If the element is underspecified, its formal component contains the formal restrictions that constrain the choice of the filler, such as a specific (set of) lemma(ta), root(s), or syntactic form(s) (e.g., part of speech). Notice that lemmata, roots and syntactic forms are themselves constructions and, therefore, refer to resources of the type `rcxn:Construction`. This is one of the ways constructions can be related to each other.

The description of construction elements goes beyond their formal components. It may also include their syntactic function, semantic contribution, and semantic properties. The syntactic function (the property `rcxn:hasSyntacticFunction`) is represented as a literal text object.[2] For the semantic contribution, contributors can either select from the predefined semantic roles in the OLiA vocabulary or provide a literal text object. The semantic properties (the property `rcxn:hasSemanticProperty`) are also represented as literal text objects. Additionally, contrib-

this paper.

[2]This is at least the current state. Experience shows that the contributors use a limited set of syntactic functions (e.g., subject, object). We may therefore introduce them as RDF resources in the future.

utors can freely add comments to include any other information about the slots not already covered by the ontology. They are represented as literal text objects of the property `RDFS:comment`.

3.2 Illustrating with examples

Constructions are furthermore linked to examples. Two versions of the example text are stored in the database: one with annotations for construction elements and one without. In both cases, the text of the example is of the type literal object. The annotation indicates which part of the text corresponds to which construction element. But we also store a version of the text without this annotation so that the text can be searched.

3.3 Linking constructions

The way constructions are organized and linked to each other is a very important strand of research in Construction Grammar. We developed for this a fine-grained typology of properties linking constructions to each other, which is discussed in Winckel (2025).

Most of the research concentrates on language-internal relationships. Paradigmatic relationships are central in the scholarly discussion, with two main types of links: vertical links and horizontal links. Our ontology defines further links to capture polysemy and metaphors.

Vertical paradigmatic links relate constructions of different levels of specificity. They are inheritance links whereby more specific constructions inherit properties from more general ones. For example, the "*Cool! exclamative construction*" inherits many features from a more generic "*Adjective! exclamative construction*". The latter subsumes exclamative utterances such as *Interesting!* or *Beautiful!*.

A vast series of links are subsumed under the property of horizontal links, which captures sisterhood or mere similarity. They relate constructions with the same degree of specificity, that share at least a part of their form or function. Typically, two constructions that inherit from the same construction and have the same level of specificity are sister constructions and can therefore be linked by horizontal links. This would be the case for the "*Cool! exclamative construction*" and the "*Amazing! exclamative construction*". While we only account for language-internal vertical links, horizontal links can be language-internal or cross-linguistic. The reason is that some of our contributors work on cross-linguistic comparison, where various degrees of similarity between constructions belonging to different languages are relevant.

3.4 Language of constructions

Our RTG includes doctoral candidates working on a wide range of current and past languages, language varieties and dialects (e.g., Haitian Creole, Middle High German, Spoken American English, Literary Chinese). Rather than attempting to compile an exhaustive list of all possible languages and dialects, which would be both impractical and unnecessary, we adopted a semi-bottom-up approach. This approach is only *semi* bottom-up, as we started with ISO 639, the international standard for language codes. ISO 639 already provides a hierarchical classification of languages, organized into broader macrolanguages (e.g., "Arabic"), with subcategories for specific varieties (e.g., "Egyptian Arabic"). Each entry in the standard includes a 3-letter code and an official English name. However, the ISO standard alone is insufficient for our needs. To address this, we expanded the list by incorporating additional language varieties and subvarieties (e.g., "Cairo Arabic" under "Egyptian Arabic"), according to the expertise of our contributors. Our OWL ontology module (with the prefix `lg:`) thus includes definitions for classes (such as `lg:macrolanguage`) and properties (such as `lg:partOfLanguage` linking a construction to a language), but also specific language instances used in our RCxn. When available in the ISO norm, the IRI names for these entries are derived from the ISO codes (e.g., `lg:ara` for Arabic), and their labels follow the ISO norm. Unlike other modules in our ontology, the `lg:` module is highly specific to our RCxn and may have limited applicability to other projects.

The IRI for a construction typically follows the pattern [language code]_[name of the construction], without blank space and with special characters (such as punctuation) removed or replaced by URL-compatible characters.

When contributors create a new construction entry via the online form, they may simultaneously generate new dummy entries. For example, if they add an inheritance link from construction A to construction B, and B does not yet exist, a minimal dummy entry for B will be created (including a title, creation date, and annotator, as described above). Similarly, if they specify that one of the construction elements in A is a lemma C that does not yet

exist in the constructicon, a minimal dummy entry for C will be created. In such cases, the newly created construction is, by default, assigned to the same macrolanguage as construction A, rather than the same specific language.[3]

3.5 Documenting research

The contributors of the RCxn are also part of the knowledge graph. Each is represented as a unique resource and an instance of the class `rsrch:Member`, which is itself a subclass of `FOAF:Person`. Contributors work on specific research questions (instances of the class `rsrch:Project`, a subclass of `FOAF:Project`). Research questions lead to findings (instances of the class `rsrch:Finding`), which derive from the work they did on specific constructions. Constructions, researchers, research questions and findings are therefore all interconnected in the knowledge graph.

Beyond researchers, the RCxn documents research data. Contributors can for example include colloprofiles for lexically underspecified slots. By colloprofile, we mean the description of the different realizations available for a given construction element. Colloprofiles include frequencies, typically drawn from a corpus study. In the RDF database, a construction element is linked to its colloprofile via the property `evid:colloprofile`, which points to a collection (or "bag" in RDF terminology) of filler entries. Each filler is represented as a blank node that contains two pieces of information: the lexical item in question and its frequency count. The latter are respectively expressed through the `evid:filler` property and through the `frac:frequency` property (adopting FrAC's architecture for collostructions). The frequency value is associated with a specific corpus (whose description also aligns with FrAC's architecture).

The description of research data is currently work-in-progress and will also incorporate descriptions of experimental data in the future.

[3]The reason is that we cannot always assume that the automatically created construction belongs to the same language variety than the construction being described. Consider for example a contributor creating a new entry for the already mentioned "*Cool! exclamative construction*" in colloquial English. The contributor could indicate that it inherits from the "*Adjective! exclamative construction*", which is not necessarily colloquial English. Assigning the macrolanguage by default avoids this potential mistake.

3.6 Statistics

As of July 2026, the RCxn contains 158 constructions from 20 contributors, distributed across 10 macrolanguages. To describe these constructions, the database contains 3607 RDF triplets. The ontology contains 44 classes and 100 properties. For a regularly updated overview of these numbers, see `https://bdlweb.phil.uni-erlangen.de/RCxn/stats`.

4 The Web app

The Research Construction is available via the web application at `https://bdlweb.phil.uni-erlangen.de/RCxn/`. The website's main feature is a list of constructions, each with its own web entry. The website also includes an interactive form that allows contributors to add construction entries.

Visitors can access the list of constructions from the landing page (via the button *Enter the Research Constructicon*). It displays an exhaustive list of the construction entries in the database, including "dummy" entries, and it is organized by macrolanguage (see Section 3.4).[4] Within a macrolanguage, constructions are organized alphabetically by title. Visitors can access relevant entries by clicking on a title.

The content of each construction entry is generated from the RDF database. Relevant graphs and subgraphs are extracted, and the page is rendered dynamically, see Figure 1. As a result, any changes to the database are immediately reflected on the website.

Construction entries contain several sections (only filled if they correspond to triplets for that construction in the database):

- General (title, meaning and comments)
- Construction elements
- Examples that illustrate the construction
- Links to other constructions
- Documentation of research linked to this construction
- Metadata

[4]For a list of constructions organized by research question, see `https://bdlweb.phil.uni-erlangen.de/RCxn/app_entries/query/`

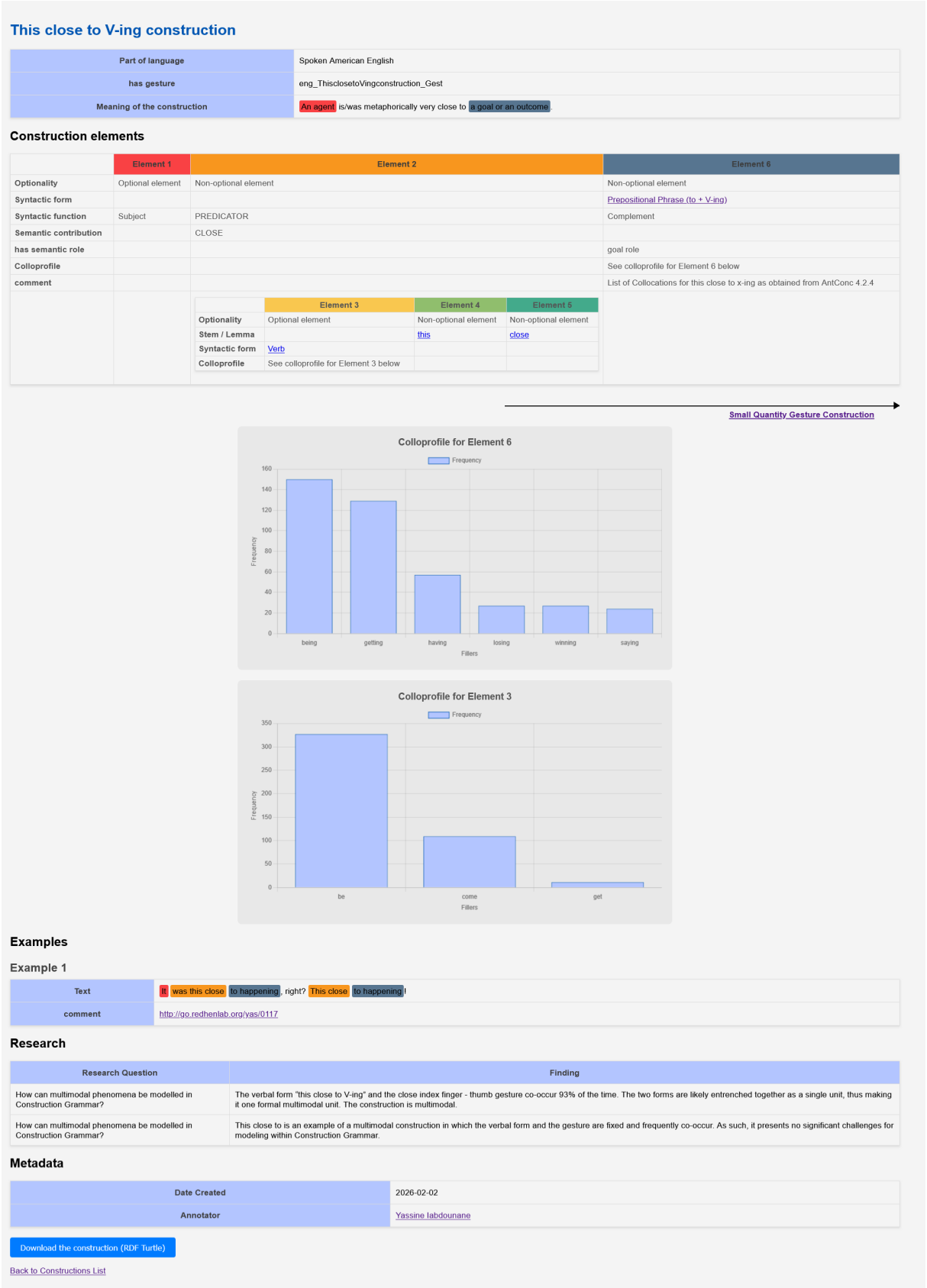

Figure 1: Screenshot of the construction entry "*This close to V-ing*" in the Research Constructicon (https://bdlweb.phil.uni-erlangen.de/RCxn/app_entries/construction/eng_ThisclosetoVingconstruction)

The display of constructions is guided by both contributors' needs and existing constructicons. For example, construction elements are presented in a tabular format, where each construction element is a column, and its features are displayed in rows, as can be seen in Figure 1. This approach is similar to the display method used by the CASA-ConstructiCon of the English Language (`https://constructicon.de/`).

Each construction element is assigned a color (red for Element 1, for example), and corresponding elements in the meaning and the text of examples are displayed in a color box of the same color on the screen. We can find similar practices in other constructicons (e.g., FrameNet Konstruktikon des Deutschen). To make the web app more accessible to color-blind individuals, a description becomes available when the text is hovered over. For instance, when the cursor hovers over text in a red box, the word "Element 1" appears in a tooltip.

In the ontologies, all resources (classes and properties) are defined with a label (`RDFS:label`) and a definition (`RDFS:comment`). This applies to the ontology developed within the project and to ontologies developed elsewhere that we use. When graphs and subgraphs are extracted, their labels and definitions are extracted as well. Labels are displayed when available, i.e., for resources, but not for literal objects. Definitions are added to tooltips so visitors can see the meaning of any resource by hovering over it.

If a construction element has a complete colloprofile (i.e., frequency numbers for a given corpus have been provided) it is displayed below the table as a bar graph.

An important aspect of a constructicon is that it is a network of constructions. This is reflected in our web interface through hyperlinks connecting constructions to each other. Below the description of construction elements and illustrative examples is a section dedicated to the types of links mentioned in Section 3.3. However, hyperlinks are also present in the table of construction elements, because, as said in Section 3.1, some elements correspond to constructions. This is visible in Figure 1 where Element 3 is described as an instance of the English "*Verb*" construction or Element 5 as an instance of the lemma "*close*".

On the webpage of each construction, visitors have the option to download a subgraph in RDF/Turtle format, containing all the triplets associated with that specific construction.

Contributors do not need any knowledge of RDF to enter a construction. A dynamic online form on the website allows them to design their own entries. Additionally, there are plans to enable editing of existing constructions in the future.

5 Conclusion

The Research Constructicon (RCxn) is fundamentally a tool designed by linguists for linguists interested in Construction Grammar (CxG). It provides a theoretically grounded, flexible, and collaborative support for constructicographic work. Though describing constructions with accuracy remains the central aspect of the constructicon, it also intends to capture the scientific process around constructions and the constructicon.

We argued that knowledge graphs are perfectly suited to model a constructicon, because they mirror cognitive models. Furthermore, Semantic Web standards combine a series of benefits: they ensure FAIR principles, provide enough formalism to bring the research on linguistics in our RTG under a common roof, while being flexible and potentially adaptable to any need that may arise. The web application furthermore offers a frontend designed to be accessible to all linguists regardless of their technical background.

The project is still work-in-progress and will continue to evolve to adapt the need of the doctoral students on the RTG. In a further future, we wish to also include contributions from researchers outside of the RTG. In the meanwhile, the current state is a proof of concept that such an endeavor is possible.

6 Acknowledgments

This research was funded by the German Research Foundation (DFG) as part of the Research Training *Group Dimensions of Constructional Space* (project no. 468527017).

We are deeply grateful to the members of the Research Training Group for their contributions, discussions, and commitment throughout this project. Our work was strongly inspired by the constructicography community, their pioneering efforts and collaborative spirit. We also extend our sincere appreciation to the three anonymous reviewers for their constructive feedback.

References

Radovan Bast, Anna Endresen, Laura A. Janda, Marianne Lund, Olga Lyashevskaya, Daria Mordashova,

Tore Nesset, Ekaterina Rakhilina, Francis M. Tyers, and Valentina Zhukova. 2021. The Russian Constructicon: An electronic database of the Russian grammatical constructions.

Carl Boettiger. 2018. *rdflib: A high level wrapper around the redland package for common rdf applications*.

Bert Cappelle. 2006. Particle placement and the case for "allostructions". *Constructions*, Special volume 1.

Christian Chiarcos, Elena-Simona Apostol, Besim Kabashi, and Ciprian-Octavian Truică. 2022. Modelling Frequency, Attestation, and Corpus-Based Information with OntoLex-FrAC. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4018–4027. International Committee on Computational Linguistics.

Christian Chiarcos and Maria Sukhareva. 2015. OLiA – Ontologies of Linguistic Annotation. *Semantic Web*, 6(4):379–386.

William Croft. 2022. *Morphosyntax: Constructions of the World's Languages*. Cambridge University Press, Cambridge.

Holger Diessel. 2019. *The Grammar Network: How Linguistic Structure Is Shaped by Language Use*. Cambridge University Press, Cambridge.

Holger Diessel. 2023. *The Constructicon: Taxonomies and Networks*. Elements in Construction Grammar.

Anna Endresen and Olaf Mikkelsen. 2024. Ten good reasons to build the Norwegian ConstructiCon, a dictionary of grammar. *Lexicographica*, 40(1):121–145.

Charles J. Fillmore. 1989. Grammatical Construction Theory and the Familiar Dichotomies. In Rainer Dietrich and Carl Graumann, editors, *Language Processing in Social Context*, number 54 in North-Holland Linguistic Series: Linguistic Variations, pages 17–38. Elsevier.

FKD. FrameNet-Konstruktikon des Deutschen. `https://framenet-constructicon.hhu.de` (accessed 19.02.2026).

Adele E. Goldberg. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. University of Chicago Press.

Miguel Grinberg. 2018. *Flask web development: developing web applications with python*. " O'Reilly Media, Inc.".

Thomas Herbst. 2019. Constructicons – a new type of reference work? *Lexicographica*, 35(2019):3–14.

Thomas Herbst and Thomas Hoffmann. 2024. *A Construction Grammar of the English Language: CASA – a Constructionist Approach to Syntactic Analysis*. John Benjamins Publishing Company.

Thomas Herbst, Thomas Hoffmann, Peter Uhrig, Armine Garibyan, and Stephanie Evert. 2023–. CASA ConstructiCon of the English language.

Thomas Hoffmann. 2019. *English Comparative Correlatives: Diachronic and Synchronic Variation at the Lexicon–Syntax Interface*. Studies in English Language. Cambridge University Press, Cambridge.

Alison Langmead, Jessica M. Otis, Christopher N. Warren, Scott B. Weingart, and Lisa D. Zilinksi. 2016. Towards Interoperable Network Ontologies for the Digital Humanities. *International Journal of Humanities and Arts Computing*, 10(1):22–35.

Arthur Lorenzi, Peter Ljunglöf, Ben Lyngfelt, Tiago Timponi Torrent, William Croft, Alexander Ziem, Nina Böbel, Linnéa Bäckström, Peter Uhrig, and Ely E Matos. 2024. MoCCA: A Model of Comparative Concepts for Aligning Constructicons. In *Proceedings of the 20th Joint ACL - ISO Workshop on Interoperable Semantic Annotation @ LREC-COLING 2024*, pages 93–98, Torino, Italia. ELRA and ICCL.

Benjamin Lyngfelt. 2018. Chapter 1. Introduction. In Benjamin Lyngfelt, Lars Borin, Kyoko Ohara, and Tiago Timponi Torrent, editors, *Constructicography: Constructicon development across languages*, Constructional Approaches to Language, pages 1–18. John Benjamins, Amsterdam and Philadelphia.

Benjamin Lyngfelt, Lars Borin, Kyoko Ohara, and Tiago Timponi Torrent, editors. 2018a. *Constructicography: Constructicon development across languages*. Number 22 in Constructional Approaches to Language. John Benjamins, Amsterdam and Philadelphia.

Benjamin Lyngfelt, Linnéa Bäckström, Lars Borin, Anna Ehrlemark, and Rudolf Rydstedt. 2018b. Constructicography at work: Theory meets practice in the Swedish constructicon. In Benjamin Lyngfelt, Lars Borin, Kyoko Ohara, and Tiago Timponi Torrent, editors, *Constructicography: Constructicon development across languages*, Constructional Approaches to Language, pages 41–106. John Benjamins Publishing Company.

Mark A. Musen. 2015. The Protégé Project: A Look Back and a Look Forward. *AI matters*, 1(4):4–12.

Yuliia Palii, Anna Endresen, Laura A. Janda, and Zoia Butenko. 2023. A Constructicon for Ukrainian: Objectives, Strategies, Results.

Malin Patel. 2026. *Corpus Evidence for Delineating Constructions*. unpublished ph.D. Thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg.

Florent Perek and Amanda L. Patten. 2019. Towards an English Constructicon using patterns and frames. *International Journal of Corpus Linguistics*, 24(3):354–384.

Zhongzhi Shi, Mingkai Dong, Yuncheng Jiang, and Haijun Zhang. 2005. A logical foundation for the semantic Web. *Science in China Series F: Information Sciences*, 48(2):161–178.

Veronika Stampfer. 2024. The development of the Evaluative-Attribution-Construction in Modern English. 10th International Conference of the German Cognitive Linguistics Association (GCLA/Deutsche Gesellschaft für kognitive Linguistik – DGKL).

Tiago Timponi Torrent, Ely Edison da Silva Matos, Ludmila Meireles Lage, Adrieli Laviola, Tatiane da Silva Tavares, Vânia Gomes de Almeida, and Natália Sathler Sigiliano. 2018. Towards continuity between the lexicon and the constructicon in FrameNet Brasil. In Benjamin Lyngfelt, Lars Borin, Kyoko Ohara, and Tiago Timponi Torrent, editors, *Constructicography: Constructicon development across languages*, pages 107–140. John Benjamins Publishing Company.

Peter Uhrig. 2015. Why the Principle of No Synonymy is Overrated. *Zeitschrift für Anglistik und Amerikanistik*, 63(3):323–337.

Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Tim Clark, Mercè Crosas, Ingrid Dillo, Olivier Dumon, Scott Edmunds, Chris T. Evelo, Richard Finkers, and 34 others. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, 3(1):160018.

Elodie Winckel. 2025. Defining relationships in the constructional network: A Semantic Web ontology for Construction Grammar. *Lexicographica*, 41(1):299–317.

Elodie Winckel, Peter Uhrig, and Stephanie Evert. 2026. A Linguistic Ontology for Constructicography: The Research Constructicon and its Ontology Modules. In *Proceedings of the 10th Workshop on Linked Data in Linguistics (LDL-2026) @ LREC 2026*, pages 87–96, Palma, Mallorca, Spain.

Alexander Ziem and Johanna Flick. 2019. Constructicography at work: implementation and application of the German Constructicon. *Yearbook of the German Cognitive Linguistics Association*, 7(1):201–214.