Evoke: exploring and extending lexicographic resources using a linked data approach

Sander Stolk
Centre for the Arts in Society, Leiden University, The Netherlands
E-mail: s.s.stolk@hum.leidenuniv.nl

Abstract. Lexicographic resources such as thesauri contain a wealth of information for research, but their published forms not uncommonly limit the ways in which users can interact with them. The web application Evoke offers users functionality for viewing, navigating, extending, and analysing content of topical thesauri. Its use of linked data mechanisms and a novel architecture (relying on the use of data catalogues, internet browser storage, annotation of URIs) addresses one of the more intractable problems in modern lexicography: allowing users to engage more fully with published lexicographic content without them infringing on licenses or requiring additional hosting. Users of Evoke can engage with lexicographic content by annotating, adding tags, and building custom queries without the need to have full, unlimited access to the entire dataset of the lexicographic resource.

Evoke is one of the first applications that provides a user-friendly interface for working with Linguistic Linked Data resources, opening these resources up to users who may not have advanced knowledge of linked data and RDF technologies. Students as well as established researchers have confirmed the user interface and underlying architecture of Evoke to be of value in answering novel research questions relying on the data of these lexical treasure troves – complemented by their own.

Keywords: Evoke, web application, software, lexicography, thesaurus, Linguistic Linked Data, linked data, data catalogue, browser storage, license

1. Introduction

Lexicographic resources such as dictionaries and thesauri contain a wealth of information for research. As Manfred Görlach points out, such resources allow one to explore not just language but also the culture of a language community.[1] They offer a lens through which users can learn many details on words in a given vocabulary, including (but not limited to) nuances in meaning, part of speech, and restrictions in regional or temporal use.

Reviews and research show that, however well-received a lexicographic work may be, it is common to nevertheless find its published form subject to the following two issues. Firstly, users are unable to query and reuse the information contained within in a way other than its editors (and publishers) had foreseen.[2–5] The physical, rigid structure of paper editions forestalls rearranging or analysing their content, of course, but even those resources that have been made available in an electronic environment are often limited to a basic set of predefined queries (e.g., browsing, searching). Secondly, researchers desire more information per lexical item than that already present. Examples of such information is whether a word or meaning is found in a particular text, or context, or is notable in some qualitative or quantitative way.[1, 3, 5–7]

It is evident that lexicographers and publishers cannot be expected to add information and functionality to fulfil all requests users may have for their individual contexts and needs. Scholars would benefit, then, from being able to add and view their own data alongside the content from a lexicographic work – the digital equivalent of scribbling in the margins of a paper edition copy. Licensing concerns of publishers, however, often restrict the ways in which users can interact with lexicographic works available online. Full access to the entire underlying database is not always made available. Moreover, there is a cost attached to hosting and moderating user
annotations. The two issues mentioned, then, are not easily resolved. Indeed, they are considered to be some of the “more intractable problems in modern lexicography”.¹

This article describes the web application Evoke, which adopts a novel approach to address the aforementioned concerns for published lexicographic works.² The application offers functionality to view, navigate, extend, and analyse topical thesauri that have been made available as Linguistic Linked Data. By employing linked data mechanisms, Evoke navigates the interests of both users and publishers of lexicographic resources. This approach aims to reduce barriers for users to start using valuable lexicographic resources and extending them – without infringing on more restrictive licenses or demanding additional data storage from the publisher of the resource.

The structure of this article is as follows. Section 2 discusses related work. Section 3 describes the web application Evoke: its architecture, use of Linguistic Linked Data, data catalogues, and internet browser storage. Before the conclusion, section 4 details ongoing research that demonstrates the impact and usefulness of Evoke and its design.

2. Related work

Linked data plays “an increasing role in eLexicography”, adopting the Linguistic Linked Data paradigm and relying on the OntoLex-Lemon model.³ Indeed, several recent initiatives aim at building and maintaining Linguistic Linked Data resources, including the H2020 projects ELEXIS (2018-22), Prêt-à-LLOD (2019-22) and the COST Action Nexus Linguarum (2019-23). Tooling in these initiatives that work with Linguistic Linked Data focus on creation, discovery, transformation, and linking.³ Cases in point are the tools LingHub⁴, which offers discovery of language resources by searching through their metadata; and NAISC³, used for aligning two RDF datasets. Unfortunately, most applications currently available for working with Linguistic Linked Data “come with a considerable entry barrier and they address the advanced user of RDF technologies rather than a typical linguist”.⁵ Evoke is one of the first applications, then, that aims to provide a user-friendly interface for such resources and to open them up to a wider audience.

Ways to extend resources on the Web by using linked data mechanisms (not specific to Linguistic Linked Data) are pivotal in notable recent work such as the tool hypothes.is⁴, used specifically for annotating webpages, and the ecosystem SOLID⁵, which relies on personal RDF data hubs. Both works use online databases to store information of users, requiring them to login to their account before they can add data. In contrast, Evoke demands no login as user data is stored locally, i.e., in memory of the internet browser, instead. The alternative Evoke offers grants users complete control over their own data and annotations (backup, share, publish), does not demand for that data to be stored online, and requires no account details before interacting with a resource and extending it. This approach both avoids public comments cluttering webpages of annotated resources and encourages users to engage in open science.

3. Evoke

The web application Evoke allows users to interact with topical thesauri expressed as Linguistic Linked Data. A topical thesaurus is “a work of lexicographical reference which presents lexical facts with semantic domains as its core organizational principle, rather than in alphabetical arrangement”.⁶ Users of Evoke, then, can explore these hierarchies of semantic domains (or concepts) and view lexical facts of words positioned in such a hierarchy. Development of the features necessary for exploring such content has been informed by feedback from both researchers and students (gathered through workshops, courses, and questionnaires) since the release of the first prototype of Evoke in 2018. The aim of the application is to allow scholars to explore, extend, and analyse linguistic data captured in a thesaurus – optionally in combination with other datasets – with ease and powered by the decentralized mechanisms that linked data offers.

The application allows users to select from a catalogue which datasets they wish to activate and ¹Marc Alexander, Director of The Historical Thesaurus of English[8], personal communication, September 28, 2020.
³https://github.com/insight-centre/naisc
⁴http://hypothes.is/
⁵https://solidproject.org/
explore (which will be discussed in-depth in section 3.3). Users can view the content of activated datasets in the browse section of Evoke, which allows them to follow any data links within or between the datasets. Thesaurus concepts from a semantic hierarchy are shown by means of a tree (see Figure 2). Clicking on a branch allows the user to dive deeper into the hierarchy. By holding down the mouse instead, or clicking on the word ‘open’ next to a branch, users can open up and view the details of the semantic concept selected (see Figure 3 for ‘The Voice’). The default tab lists the various words that express the meaning of the concept in question. Other tabs available provide information on the identification of the concept (e.g., URI and name), statistics on the distribution of the words within this taxonomy branch, and word clouds.

Lexical facts of specific words are available, too, and can be accessed by clicking on a word listed. The default tab then provides the identification of both the word in this specific sense and the word in general, the part of speech, synonyms, tags, and annotations (see Figure 4 for the word ‘thief’ in the sense of “A robber, thief”). The other tabs in this view offer all known meanings of the word as a list or as a word cloud. Users can add their own annotations for any resource they view, which will be discussed in-depth in section 3.4.

Next to browsing, Evoke can be used to perform more advanced analyses over the data available. The statistics section of the application offers the possibility to build custom queries for analysing the use of specific tags, parts of speech, and/or languages (see Figure 5). A selection can be contrasted with all words (default), other parts of speech, other languages (if multiple are present), or another custom selection of features. Analysis results for a query are shown in the form of charts that convey the item count of a selection, its degree of ambiguity (indicating polysemy), its degree of synonymy, its distribution over taxonomy subcategories (see Figure 6), and its distribution over the depth of the taxonomy (indicating how specific in meaning the selected words are; see Figure 7).

Finally, the user section of Evoke allows users to make a backup of their annotations, reactivate backups, discard their user data, or publish their data to an online data service.
3.1. Architecture

Evoke employs client-side rendering, which entails that the user’s internet browser does not fetch entire webpages from a server but fetches only data necessary to fill out pages that it itself composes. Use of this rendering technique has the advantage of reducing the amount of resources required server-side. The interface of Evoke is rendered using React and Reactstrap (basic HTML) complemented by libraries specifically intended for vector graphics (i.e., D3, D3Pie, Recharts, and Wordcloud).6

The application offers a high degree of flexibility in managing which content is to be viewed or analysed, including where individual datasets have been made available. The functionality to realize this hinges on the use of so-called data catalogues, which lists datasets available for querying. Upon start-up, Evoke loads a default data catalogue. Custom data catalogues can, however, be used too. The use of data catalogues in Evoke will be discussed in more detail in section 3.3.

Datasets listed in a catalogue are accessed through Ajax calls to SPARQL endpoints and/or APIs. As Ajax calls to data services are asynchronous, the Evoke web application contains a so-called high-order component that acts as loader. This loader wraps the desired component in one that first awaits data requests. During the loading time, a loading icon is presented. Once the data required has been retrieved, the loader renders the wrapped component with its proper input. This mechanism is used, in combination with the underlying data form of Linguistic Linked Data, to follow links iteratively in fetching further information from all active datasets. Thus, when a user selects a semantic concept of a thesaurus for viewing, a list of words that express that concept is retrieved. For each of those words, their URI is used to retrieve all associated tags available across the active datasets – whether part of the original thesaurus data or a set of annotations created by others.

The application ingests, next to data served remotely via SPARQL or API calls, linked data stored in the internet browser of the user. A dataset in the browser can be stored using either the Turtle serialization or in JSON-LD and is interpreted using libraries from the Comunica framework.[13]7 The use of this storage method is detailed in section 3.4.

3.2. Linguistic Linked Data

Evoke adopts the Linguistic Linked Data paradigm for the way it accesses, explores, and extends data. This paradigm builds on linked data principles, which advocate the use of Web mechanisms for capturing and sharing data, employing graph-based models (i.e., nodes and relations between them) and identifying data by means of URIs (often Web addresses).[14] The use of URIs allows one to capture and identify data in a language-independent manner, reuse terminology defined elsewhere, and to create links between datasets or nodes within different datasets.

Applying linked data principles to language resources nets a number of benefits.[15] One of these benefits is that its data form enables the merging of datasets in order to obtain a valid combined set of data. Thus, linguistic resources and datasets elaborating on them can be queried in unison. A second benefit is an increased level of interoperability. Using standardized terminology in describing linguistic data increases a shared understanding of that data and facilitates their interpretation by software. Moreover, the use of URIs as identifiers ensures data can be linked without the need for duplication of information from one dataset into the other. The ability to link (or reference) in such a manner is valuable in the setting of Linguistic Linked Data, since it is not uncommon to find lexicographic resources subject to licenses meant for viewing only and stipulating users are not allowed to copy or download a substantial portion of its content (e.g., [8, 16]). By adopting URIs in published lexicographic resources, then, their users can start exploring and extending these resources at an early stage and may be enticed to engage more fully with the content offered without infringing on such licenses.

The Evoke web application, in order to draw on the aforementioned benefits, assumes language resources to adopt Linguistic Linked Data as specified by the W3C OntoLex community,[17] applied specifically to the context of topical thesauri.[18] The use of linked data in Evoke is, however, not limited to solely the content of datasets catalogued. The data catalogues

---


7The Comunica library n3 is used for Turtle; jsonld-streaming-parser for JSON-LD; actor-init-sparql-rdfjs for SPARQL queries in browser memory.
themselves and offline data stored in the internet browser employ this data form, as will be discussed in the next sections.

### 3.3. Data catalogue

The data catalogues in Evoke adhere to the W3C DCAT vocabulary, an international standard specifically for expressing datasets and the services that provide these sets, including access details.[19] This information is stored in the JSON-LD format and can therefore be read as JSON or, through the context provided, interpreted as RDF.[20] The use of these standards are meant to accommodate a higher level of interoperability with other tooling and services. An example catalogue is shown in Listing 1. Users can add their own datasets or data services to a catalogue, store their catalogue locally as a JSON file, and activate a local catalogue by dragging and dropping it onto the Evoke web application.

Access to datasets may or may not need to be limited, depending on the usage license associated with them. In order to ensure that Evoke can work with more restrictive licenses, too, two types of access mechanisms for data services are supported in data catalogues: (1) a SPARQL endpoint and (2) the Evoke API. The former allows services to respond to any query using the standardized querying language for RDF. The latter ensures content can be viewed and browsed in Evoke through a basic set of queries specific for this need, without offering users full access (that is, the possibility to extract or download the full dataset). Distinguishing between these two types of access mechanism in the data catalogue is achieved through different values for the `endpointDescription` attribute of data services.[8]

Which datasets listed in the active catalogue will be queried by Evoke depends on the selection made by the user. The Evoke interface allows users to enable (or disable) datasets listed. Only those datasets can be enabled that (1) have a data service associated with them and (2) already have all of their required dependencies enabled. To illustrate, the ‘Riddle 47’ dataset contains links to the dataset ‘A Thesaurus of Old English’ and depends on it for analyses. Once a user has enabled this required dataset in the interface, that user can opt to enable the ‘Riddle 47’ dataset, too (see the top bars in Figure 4). Datasets served by the same service, though available in different graphs, are queried in unison and allow statistical analyses to be performed.9

### 3.4. Browser storage

Any user of the Evoke web application can annotate linguistic content, such as words or semantic concepts, with information relevant to them. Typing a sentence in the annotation component of a page will automatically create a linked data annotation that adheres to the Web Annotation standard of W3C including any extracted tag when a hashtag is used (see “#riddle47” in Figure 4). The novel aspect of this approach is that such an annotation is not stored in an online database but is instead stored locally in the user’s internet browser, employing the `localStorage` attribute of Web storage.[22] Annotations stored in the browser can be downloaded as a file to backup and can be reactivated in the browser – providing users with full control over their created content and allowing them to share it in the manner of their choosing (see Listing 2). Publishers of the original lexicographic resource benefit from this approach, too, as they neither need to moderate, store, or host annotations nor offer users login mechanisms before they can interact with the information. Costs for hosts may thus be substantially reduced for presenting users with this functionality.

Annotations contain references to the identifiers, or URIs, of the original lexicographic content without including the raw data of the annotated content in the annotation itself. This approach can entice users to already explore dictionaries and interact with them, formulate a plan of research, and at a later stage take the hurdle in getting support for further research from the publishers – be it in the form of a more open license, getting access to advanced services, or asking assistance from the expertise of lexicographers. Users can be said to have an invested interest in the lexicographic resource at this point. Moreover, their additions are explicit, digital, and can be used in this

---

[8]Evoke does not perform statistical analyses over data found at different data services, since that would require assembling a full picture of the relevant data either at the user, one of the data services, or a data aggregator (be it query results or parts of datasets), which is an intricate process, time-consuming, heavy on network traffic, and possibly restricted or made impossible by the licenses that are applicable to the datasets.
form for analyses when queried in unison with the original dataset (facilitated by the characteristics of Linguistic Linked Data).

User data stored in the internet browser can, as with any RDF dataset, be published to a data service and added to a data catalogue for use in Evoke. Indeed, when one publishes through the Evoke user interface, a new data catalogue is created automatically in which the published dataset is listed. This updated catalogue is not made public by the application: as with other user content, the catalogue is stored in the browser. A means to download the updated catalogue is provided to the user immediately after a successful publication. Users can choose to share it with others in a way they see fit or, if they desire to share the newly published dataset publicly, upload it to a public server and/or contact the administrator of the application to request inclusion of the dataset in the default catalogue.

4. Case study: A Thesaurus of Old English

To explore the impact of Evoke and the solution suggested, a research project has been organized for a case study of a single thesaurus. This project has brought together 17 scholars from 8 universities and lexicographic institutions from several countries to explore the contents of A Thesaurus of Old English using the application.[23] This thesaurus, transformed to Linguistic Linked Data,[24] captures the lexis of the early medieval variant of English that was spoken between roughly 500 and 1100 AD. Each investigation within the research project explores the lexis from the perspective of a certain discipline: linguistics, literary-criticism, history, lexicography, philology. In their explorations, the researchers (and, in some cases, students) set about in viewing, adding to, and analysing content of A Thesaurus of Old English. The majority of the researchers employ the annotation system of Evoke to add relevant data. Some, however, utilize linked data mechanisms and link datasets through other software first before viewing and analysing the results in Evoke.

Examples of research done within this project are linking up words (or word senses, rather) from Old Frisian and Old Dutch to the thesaurus taxonomy. Doing so will allow researchers to contrast how many nuances these language communities respectively were known to have, next to Old English, in expressing certain concepts such as kinship or greetings. Other research focuses on adding information on the occurrence of words in certain medieval texts (e.g., Beowulf) or their use by a certain author (e.g., Ælfric), offering insights into the onomasiological profiles associated with authors or genres. These profiles may act as semantic fingerprints in identifying authorship or in positioning other work. The outcomes will be presented at the pre-conference EASE workshop of the 21st edition of the International Conference of English Historical Linguistics (Leiden, June 2021)10. Its proceedings will be published in a special issue journal of the Amsterdamer Beiträge zur älteren Germanistik11 (Fall 2021). Preliminary results show that both students and established researchers – including ones who had no prior knowledge of linked data and RDF technology – are able to utilize Evoke in answering novel research questions.

5. Conclusion

Evoke is one of the first applications that provides a user-friendly interface for working with Linguistic Linked Data resources, opening these resources up to users who may not have advanced knowledge of linked data and RDF technologies. The web application employs linked data mechanisms to offer functionality for viewing, navigating, extending, and analysing content of topical thesauri together with relevant, linked datasets. The novel architecture of Evoke employs elements – i.e., data catalogues, internet browser storage, annotation on URIs – that address some of the more intractable problems in modern lexicography: licensing and hosting concerns that prevent offering functionality to users to interact with the lexicographic resources they value. Evoke demonstrates that, by employing linked data mechanisms, interests of both users and publishers can be taken into account so that users can extend these resources and formulate new queries – without infringing on licenses or demanding additional data storage from the publisher of the resource. The web application Evoke, then, allows for a higher engagement of users with published lexicographic resources in exploring and extending these lexical treasure troves.

Future work on Evoke will aim to incorporate functionality most frequently requested by

---

10https://www.universiteitleiden.nl/icehl21
11http://brill.com/abag
researchers. These include the ability for users to hide content (e.g., filtering based on user-defined tags). This feature would effectively allow users to create subthesauri. A second helpful feature is one that indicates which datasets from a data catalogue contains statements on a specific URI (be it a word, word sense, semantic concept, or anything else). Thus, Evoke should be able to foreground which datasets may be relevant for a user’s context. Additionally, such a feature will be helpful to locate works that have engaged with one’s own.

References


Fig. 2. Evoke showing the taxonomy of ‘A Thesaurus of Old English’.

Fig. 3. Evoke listing words denoting “The voice” from ‘A Thesaurus of Old English’.
Fig. 4. Evoke showing a sense of the word *þeof* as described in two datasets (i.e., ‘TOE’ and ‘riddle47’).

Fig. 5. Evoke showing the form that allows users to select their features of interest in the statistics section.
Fig. 6. Evoke showing the distribution over semantic concepts of word senses labelled ‘riddle47’ (orange) versus all senses (blue).

Fig. 7. Evoke showing the distribution over taxonomy depth of word senses labelled ‘riddle47’ (orange) versus all senses (blue).
S. Stolk / Evoke


Listing 2 User-created annotations of two word senses, which are tagged to indicate they occur in the Old English text *Exeter Riddle 47.*