International Journal of Lexicography, 2024, **XX**, 1–27
https://doi.org/10.1093/ijl/ecae001
Advance access publication 29 March 2024
Article

OXFORD

A philologist's perspective on Artificial Intelligence – a case study into *English Dialect Dictionary Online 4.0*

Manfred Markus[1] and Monika Kirner-Ludwig[2]

[1]English Department, University of Innsbruck, Austria (manfred.markus@uibk.ac.at)
[2]English Department, University of Innsbruck, Austria (Monika.Kirner-Ludwig@uibk.ac.at)

Abstract

Our paper seeks to bridge the gap between a linguistic and a technological description of Artificial Intelligence, as we see chatbots and other representations pushing to the forefront of academic ways of working. We set out to venture a philologist's perspective on AI, critically reflecting upon the extent and nature of a lexicographical software's 'intelligence.' We shall discuss and demonstrate AI's potentials and limitations by adhering to the specific case of the recently released *English Dialect Dictionary Online 4.0* (Markus and team 2023). While we propose that *EDD Online* is 'intelligent' in the sense that its retrieval software will anticipate and be capable of answering many of the questions that dialectologists would raise, we claim that the reason for such 'intelligence' is inherently rooted within the competence of humans in terms of predicting such questions and of eliciting the answers by means of complex query rules. We shall in fact argue that – as opposed to retrieval software types like the hotly debated ChatGPT chatbot, which are merely black box phenomena – *EDD Online 4.0*'s user interface represents a rare feature of AI: transparency and proactive user-friendliness. (185 words)

Keywords: dialectology; lexicography; Artificial Intelligence; *English Dialect Dictionary Online*; corpus linguistics.

1. Introduction

The inner build-up or processing steps facilitated by modern software programs – even though put to use efficiently and widely – mostly remain a black box to the average philologist. In order to describe what most of us will leave to IT-experts to take care of, we tend to take metaphorical terms such as "training" and "deep learning" at their face value and at the same time admire yet tend to avoid this 'new world' of Artificial Intelligence (AI).

For many, the term *Artificial Intelligence* (AI) may call to mind automated tools and self-driving cars. However, the notion of *AI* implies complex computer programming applied not only to machines but also to communication and information strategies. This is an issue that has recently been peaking in collective saliency through the launch of ChatGPT ('Chat Generative Pretrained Transformer') by OpenAI in 2022, which demonstrably illustrates the capability of creating conversational text not only at a user's choice of length and format, but strikingly so at their preferred style and desired level of detail; it essentially integrates and makes use of AI as a conversational agent, mimicking human-human communication.

While a number of recent scholarly works across disciplines have been discussing the potentials and risks of ChatGPT as an artificial intelligence chatbot (e.g. Carlbring et al. 2023; Hill-Yardin et al. 2023; Yu 2023), the angle this paper takes has, to our knowledge, not been chosen before, that is the philologist's perspective. We suggest describing the new version 4.0 of *English Dialect Dictionary Online*, an electronic lexicographical tool and rich dataset, in terms of AI, which means that we will focus on the algorithms it was trained on in order to facilitate the multifarious search routines available to the user of *EDD Online*. Its interface as created by our team of researchers at Innsbruck is much more than a digitised version of a dictionary. It is a computerised query platform tailored to apply a large number of search parameters and filters to Joseph Wright's *English Dialect Dictionary* (*EDD*; 1898-1905), which, with its 4,670 pages in six volumes, is the most comprehensive English dialect dictionary ever published.

The interface of the new *EDD Online* platform 4.0 (Markus and team 2023) can claim to be based on 'intelligent' software in that it provides answers to questions that no dialectologist might have raised, let alone have been able to answer before. In order to make our argument in favour of AI being applicable to *EDD 4.0,* we put forth the following sections in this paper: Section 2 provides a short overview of AI. Section 3, after a brief introduction to *EDD 4.0*, describes its 'intelligent' retrieval strategies by selecting the processor BaseX and its query language XQuery as an example of software usage. Section 4 provides the 'interface' of *EDD Online 4.0*, that is, the surface based on previous programming and created by web design; this is the amenable mode of information users are faced with. Section 5, abiding by the interface surface, sheds light on 'intelligent' retrieval strategies from the point of view of dialectology. In three sub-sections we will, first, emphasise the great value of a comprehensive or 'synthetic' approach in a modern kind of dialectology; second, introduce and illustrate the so-called 'combinatorial trap' in the application of data; and, third, use the filter of usage labels as an example of retrieving valuable results from complex data. The last analytic section of the paper familiarises the reader with a particularly remarkable aspect of artificial competence in *EDD Online*, the *ad hoc* creation of virtual maps.

The purpose of the present paper is neither to familiarize the reader with the diverse tools of *EDD Online* (for this see the *Guide* in the interface itself and Markus 2021a) nor to present an AI-tutorial 'for dummies,' but to help tear down the barrier between (corpus) linguistics/lexicography, on the one hand, and intelligent programming, on the other. As such, this paper is clearly intended to target an interdisciplinary approach for philologist readers. This is a point we wish to stress before we dive into sketching out our applied understanding of the notion of *Artificial Intelligence*.

2. Some philologically relevant thoughts on Artificial Intelligence

Being the fashionable and notorious term it is,[1] any definition of AI will essentially depend on the scholar's perspective and the context of usage. Recent and present definitions tend to focus on a software's capability to use data for independent 'learning,' in particular in the form of 'Deep Learning' and 'Neural Networks.' However, the resulting autonomous generation of sensible text and speech and the identification of hitherto unknown images are only the last phase in the development of AI software over the last decades. In the general and less recent sense of the term, AI is concerned with getting computers to do tasks that would normally require human intelligence (Raphael 1976). The definition from the CIRP Encyclopedia of Production Engineering (2014) sounds similar: '[AI] is the science and engineering of making intelligent machines, especially intelligent computer programs that exhibit characteristics associated with intelligence in human behaviour'.

As early as 1950, Turing, in his famous visionary article 'Computing Machinery and Intelligence', after a tentative definition of 'digital computers', warned us of the 'danger of circularity of argument' (Turing 1950: 436). Indeed, definitions of the type quoted above

may be called tautological, given that they use the tricky terms *intelligence* and *intelligent* as *definientes* though they are clearly part of the *definiendum*. Other hazy terms used in other definitions of AI are *learning* and *creativity* (Otte 2021: 33-4; du Sautoy 2021: 17-25; Herger 2020). All these notions are rather void without a wider frame of reference. Philosopher Richard David Precht tried to provide – in a well-received and also partly disliked book (2020; for a critical assessment see Kirschfeld 2020) – a human frame of reference, arguing that all those positive claims of AI-adherents (e.g. Eugster 2017, Herger 2020, and Ing & Grossman 2023[2]) are futile simply because, '[c]ontrary to computers, human beings do not think rule-based" (Precht 2020: 25).[3] From a linguistic and educational point of view, Noam Chomsky, in a 2023 interview, critically discredited ChatGPT as 'basically high-tech plagiarism' that 'undermines education.'[4]

There seems to be a deep misunderstanding between, on the one hand, the technocrats, the practitioners, serving the commercial purposes of globalised industries, the Silicon-Valley mindset, and, on the other hand, the less optimistic seers, thinkers, and philosophers, such as Precht. In our opinion, warnings coming from the latter group are justified. Chomsky is right in his harsh verdict that ChatGPT is 'tantamount to avoiding learning' (Bastian 2023[5]), given that his educational concept of 'learning' is different from that of computer scientists. Also, rationalisation of work, for example by robots, not to mention weapons (AWS), and the automatic tailoring of commerce to individual needs of consumers have far-reaching repercussions on our lives, values, and societies as a whole. If these repercussions remain non-reflected or are ignored for the sake of efficiency (of selling or influencing), we will all (or nearly all) have to 'pay' for it (to put it in a commercial metaphor). The metaphor stands for loss of control, loss of human contact, increase of cyber criminality, and reduction of education and human awareness. It is obvious that, behind the two positions, there are really two basic philosophical and political attitudes.

In this paper, however, we have to leave aside any such big questions pertaining to whether and to what extent capitalism and consumerism are looming behind AI and to whether our deeper human interests can be substituted or satisfied by computer software.[6] Instead, we claim that *EDD Online* is 'intelligent,' if we acknowledge that the software of *EDD Online* anticipates many of the questions that dialectologists would raise and that it, unlike them, is capable of answering. The deeper reason for such 'intelligence,' as we propose, is the competence of our team in terms of predicting such questions and of eliciting the answers by means of complex query rules. While part of the software's 'intelligence' consists in the computer's superhuman memory needed to retrieve the stored data according to the user's interests, its main part is the retrieval process itself, which equals looking for a needle in a haystack.

As in feed-forward neural networks, there are basically three levels[7]: the 'haystack' is the input layer, the 'needle,' that is, the retrieved data, is the output layer, and our software's algorithms that generate the output can be addressed as the intermediary 'hidden' layers. Neural nets generally have many 'hidden' layers. Figure 1 shows a model with just two of them. Applied to *EDD Online*, the data of the green input layers shown in Figure 1 could be three entries identified by a certain input string; layer 1 could address four compounds that include that string, layer 2 four selected dialect areas; the output layer would contain the list of the query results. However, beyond this example, *EDD Online* actually contains thousands of entry inputs, multiple hidden layers, and thousands of possible outputs, which means that the network at work in our case is far more complex than the model in Figure 1 suggests.

We are, of course, aware that the degree of complexity in *EDD Online* is no match for ChatGPT or similar text generating tools, as regards the quantity of layers, and that modern neural networks have branched into various subtypes over the last few years. We are also aware that special techniques concerning hidden layers such as the attribution of (statistical) 'weights,' 'punisher lines' (Shane 2021: 73), secondary ('pruned') layers (cf.

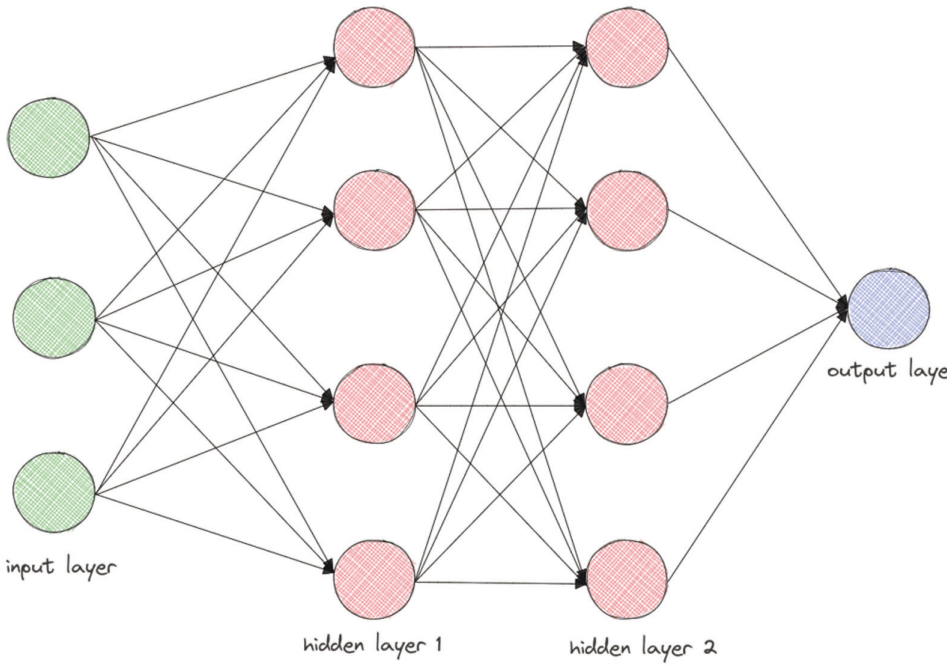

Figure 1. Model of a simple feed-forward network (borrowed from Antoniadis 2022)

Zeng & Yeung 2006), and so forth, may be part of the 'training' or 'learning' capacity of the machine. However, the greater the complexity of hidden layers, that is, of the black box, the more testing in terms of trial and error come into play. This is what visualization, translation software and ChatGPT are working with: unsatisfactory outputs lead to repair work on the 'hidden' layers. Even when this feedback is applied automatically, some human mind will still have to have programmed the automatism in the first place.

In our opinion, the difference between this methodology of neural networks and that of *EDD Online* is less striking than it may seem at first sight. In our case, the training and learning was not triggered by statistical majorities of users, but by the Innsbruck project team and its modification of the hidden layers, that is, of the search algorithms that we had previously implemented. The total of these algorithms had soon grown into such a complexity that no one in our team could keep up with having them all mentally available at any time. Our method was partly based on trial and error as well as on ensuing fine-tuning, which are also characteristics of neural networks, or a special branch of them ('Reinforcement Learning,' Frackiewicz 2023). Our refinement work on our 'hidden' layers included nearly 200 cycles of testing, with each test on our server lasting for many hours.

Summing up this section, AI, over the last few years, has reached an admirable complexity, using machine 'training,' where the computer can learn 'independently, but only within the framework that the data and algorithms provide.' This is at least the answer of an EDGE chatbot to our question whether AI can learn independently.[8] The ChatBot goes on to say that AI systems 'cannot go beyond their own context or generate creative ideas.' In other words, there is always the framework of a context. In modern AI, the context is intransparently wide, but in principle there will be a manipulated context just like in the case of *EDD Online*.

The following section will shed light on the retrieval software used in *EDD Online 4.0*.

3. Opening the black box, or: how we made AI work for *EDD Online*

3.1 A brief introduction to EDD 4.0

EDD Online is an internet platform or 'interface' based on Joseph Wright's *English Dialect Dictionary* (published 1898-1905). It covers the time from 1700 to 1904 and, spatially, both the United Kingdom and the main English-speaking countries in the world. *EDD Online* is the product of a research project carried out at the University of Innsbruck from 2006 to 2023.[9]

In line with Wright and his team's detailed description of dialect items, the interface of *EDD Online 4.0* is multifaceted and complex. It has 18 search criteria, dozens of sub-filters, and hundreds of sub-sub-filters (or search keywords). The rich information on dialects that had hitherto been hidden in the *EDD* is now accessible and retrievable by linguistic criteria. One of the main benefits of the interface is the combinability of search criteria, another the statistical quantification of search results of dialectal distribution on *ad-hoc* maps.

The structure and great potential of *EDD Online* has been described in detail in various publications (especially in Markus 2021a). Its most recent version 4.0 includes a revision of the logical potential of the interface (addition of the operator ONLY) as well as an examination of the combinability, of the thousands of sources and of the illustration on maps; for details of the innovations of 4.0 versus its predecessors see Markus (2023).

3.2 Programming 'life' into *EDD Online*

In order to illustrate the complexity of our query rules, we invite the reader to first delve into an example of a query command carried out in BaseX. This is a search and editing processor for the query language XQuery; both are needed for retrieving information out of XML databases.[10] XML ('eXtensible Markup Language') is a presentation mode of text data which keeps the text legible and focuses on precision by allowing for tagging each single string, but is less adequate for presenting the hierarchical relations between strings.

To grasp these, TEI (the 'Text-Encoding Initiative') has been invented, which is essentially a coding convention for all kinds of texts. To illustrate the difference, Figure 2 shows a short text first in XML and then in TEI.

In the XML extract in Figure 2, the strings are all sequentially presented, no matter what their rank in the syntax of the text is. In TEI, on the other hand, each category has to be opened by < and closed by/>, embedding subordinate categories, which, for their part, have again to be correctly opened and closed. So, the structure of TEI text equals that of Russian matryoshka dolls. Accordingly, in Figure 2, the 'name type' re_COMB, offered by TEI for marking a word combination, subsumes number (9) of a list of combinations and, on the same level, adds the combination concerned (*Maiden pasture*).[11] After the coding of the comma, the 'name' is closed again. The benevolent reader should ignore whatever else is presented in our extracts. Our point is that the hierarchy between the *numerus currens (9)* and the 'term' of the combination is preserved, as is the correlation with the meaning of *maiden-pasture*, in seven following lines also called 'term', but now subordinated under 'def' (for 'definition').

The strict hierarchy which manifests itself in the TEI extract in the form of specially designed abbreviations (such as *name*, *term*, and so forth) merely concerns the outer appearance or structure of the TEI text. The hierarchical relations as such are borrowed from the XML text, where they are hidden in the tag-chains and where they are partly human-made.[12] In other words: the computer 'knows' about the implicit hierarchy of the text because we tagged the text accordingly in the first place. The conversion from XML to the TEI structure, however, was carried out by an additional unambitious software in next to no time.

It would take us too far to comment on all the steps of processing necessary to retrieve data of the user's needs from the dictionary, given that so many steps are involved. We will limit our present discussion to the task of retrieving data with the help of BaseX commands. As an example, we will pretend to be searching for the dialectal string *maiden* as a unit within a word combination, as selected in Figure 2. The mask of BaseX is shown in Figure 3.

```
<re_COMB_name_COMB_span_NUM value='(9)' n='W00152'
facs='t=3116,b=3160,l=2656,r=2709' />
<lb n='W00153' facs='t=3146,b=3207,l=1563,r=2790' />
  <re_COMB_name_COMB_term value='{{Maiden}} pasture' n='W00154'
facs='t=3151,b=3193,l=1563,r=1741' rend='bold' />
  <re_COMB_name_COMB_pc value=',' n='W00155' ana='pc_prae' />
  <re_COMB_def_term value='grass' n='W00156' facs='t=3160,b=3195,l=1757,r=1868' />
  <re_COMB_def_term value='land' n='W00157' facs='t=3151,b=3186,l=1885,r=1974' />
  <re_COMB_def_term value='which' n='W00158' facs='t=3151,b=3188,l=1991,r=2119' />
  <re_COMB_def_term value='has' n='W00159' facs='t=3154,b=3189,l=2138,r=2206' />
  <re_COMB_def_term value='never' n='W00160' facs='t=3165,b=3190,l=2225,r=2345' />
  <re_COMB_def_term value='been' n='W00161' facs='t=3157,b=3192,l=2360,r=2456' />
  <re_COMB_def_term value='ploughed' n='W00162' facs='t=3158,b=3205,l=2471,r=2685' />
  <re_COMB_def_pc value=';' n='W00163' ana='pc_prae' />
-----------------------------------------------
<name type='COMB'>
      <span n='W00152' facs='t=3116,b=3160,l=2656,r=2709' type='NUM'>(9)</span>
       <lb n='W00153' facs='t=3146,b=3207,l=1563,r=2790' />
       <term n='W00154' facs='t=3151,b=3193,l=1563,r=1741'
rend='bold'>{{Maiden}} pasture</term>
       <pc n='W00155' ana='pc_prae'>,</pc>
      </name>
      <def>
       <term n='W00156' facs='t=3160,b=3195,l=1757,r=1868'>grass</term>
       <term n='W00157' facs='t=3151,b=3186,l=1885,r=1974'>land</term>
       <term n='W00158' facs='t=3151,b=3188,l=1991,r=2119'>which</term>
       <term n='W00159' facs='t=3154,b=3189,l=2138,r=2206'>has</term>
       <term n='W00160' facs='t=3165,b=3190,l=2225,r=2345'>never</term>
       <term n='W00161' facs='t=3157,b=3192,l=2360,r=2456'>been</term>
       <term n='W00162' facs='t=3158,b=3205,l=2471,r=2685'>ploughed</term>
       <pc n='W00163' ana='pc_prae'>;</pc>
      </def>
```

Figure 2. Linear XML presentation vs. hierarchical TEI presentation

The top window on the right [= (a)] is provided for filling in the search command. The bottom window on the right [= (b)] offers programming help or syntactic mistakes in the search command, and the one on the left [= (c)] presents the search result after the query has been run. In the following, we shall focus on the search command and the ensuing contents of the result box.

The BaseX search command for *maiden* in a word combination comprises 62 lines.[13] We will first consider the initial 26 lines of the command, which algorithmically define and redefine dialect criteria with the dollar sign; we call this part the 'definition of items' (a1). The rest of the command (lines 28 to 63) contains the logical core of the algorithm in view of the intended output; we will call this the 'definition of output' (a2). The result itself, shown in the left bottom window of the BaseX mask, will be commented on subsequently under (c). So, schematically, the following steps are taken (Table 1):

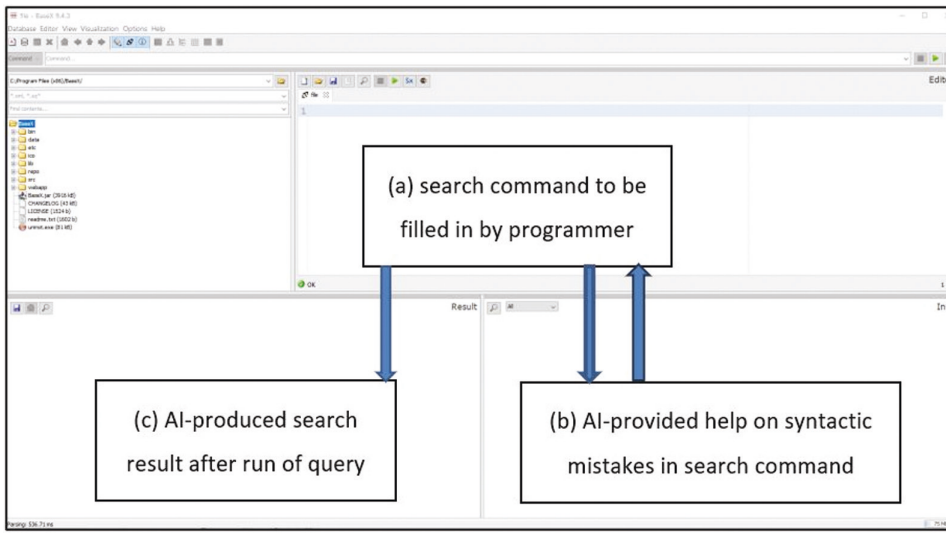

Figure 3. Mask of BaseX featuring steps (a), (b) and (c).

Table 1. Steps to be taken for query programming in BaseX

a.	Search command window		
a.1	first half	definition of items	cf. Figure 4
a.2	second half	definition of output	cf. Figure 5
[b.]	[not considered here]		
c.	Result window: presentation of result		cf. Figure 6

3.2.1 Definition of items (a1)

The beginning of the command that searches for the string *maiden* as part of *combinations* looks as shown in Figure 4.

The lines of Figure 4 contain, apart from an attribution, a number of algorithmic definitions. The very first line makes clear what string we are looking for and to what part of the text the command is meant to apply: the TEI version of the text, the tag *re* (for 'reference') with the 'attribute' *COMB*, and the string *maiden*, which is rightly tagged as *term* of a *name* (of a combination).[14] The detailed writing rules and symbols, such as slashes, double colons, brackets, and so forth, are abbreviated markers of the hierarchical position of the items at issue (see any XML handbook).[15] The *let*-commands in lines 2 to 25 are auxiliary names of text features and their definitions in the hierarchy of the text. For example, in line 3, *$entry* is the name of an entry that is an 'ancestor' of *$comb*; *$comb* had to be defined in a previous (the second) line. As one can see in the other lines, some of them are more complex. By and large, the lines contain all the items that may be of interest to the user of *EDD Online* in the context of a search for *combinations*, even though the eligible parameters and filters are not all used in a concrete query.

3.2.2 Definition of output (a2)

We skip the details and zoom in on the second half of the formula searching for *maiden* as part of a *combination*. Figure 5 shows what the software is supposed to retrieve as a *result* of the above query.

```
1  for $combTerm in (TEI//re[(@type="COMB")]/descendant::name/term) [matches(text(),"maiden","i")]
2   let $comb:=$combTerm/ancestor::re[(@type="COMB" )]
3   let $entry:=$comb/ancestor::entry
4   let $compTermNum:=$combTerm/@n
5   let $orth:=$entry/form/orth
6   let $pos:=$entry/form/*/pos
7   let $span:=$combTerm/preceding-sibling::span[1]
8   let $spanValue:=replace($span,"[()|[)]","" )
9   let $cit:=$comb/following-sibling::gb[1]/following-sibling::*[not(matches(name(),'lb'))][1][name()='dictScrap' and @change='CIT']
10  let $spanCit:=$cit/descendant::span[@type="NUM"]
11  let $spannext:=$spanCit[(replace(text(),"[()|[)]",""))= $spanValue][1]
12  let $area0:=($cit/descendant::*[((preceding-sibling::span[@type="NUM"][1]/preceding-sibling::*[1][name()='span' and @type='NUM'])
    [text() = $spannext/text()]  or ( preceding-sibling::span[@type="NUM"][1])[text() = $spannext/text()] )]  )
13  let $area1:=$comb/descendant::*[preceding-sibling::name[1]/term[text() = $combTerm][1]]
14  let $searchArea:=if(not($spanValue))  then ( $area1|$cit ) else ( $area1|$area0 )
15  let $dialect:=$searchArea/(descendant-or-self::district | descendant-or-self::region | descendant-or-self::country)
16  let $dialect3:=($comb/preceding-sibling::form[1]/(district | region | country)| $ comb/preceding-sibling::form[1]/* [not(matches(
    name(),'usg'))] / descendant-or-self::title/(district | region | country))
17
18 let $dialectExist:=( $comb|$cit)/( descendant::district | descendant::region | descendant::country )
19  let $compPos:=$entry/descendant::name[(@type="COMB")][descendant::term[(@n=$compTermNum)]]/following-sibling::gramGrp[1]/pos
20  let $etym:=$entry/dictScrap[(@change='COMMENT' )]/descendant::lang
21  let $label:=$searchArea/(descendant-or-self::note[(@type, 'LABEL')]| descendant-or-self::note[( @type='GRAM')]| descendant-or-
    self::gram  | descendant-or-self::m)
22  let $years:=$searchArea/descendant-or-self::date
23  let $prev:=string($entry/@prev)
24  let $select:=string($entry/@select)
25  let $title:=$searchArea/descendant-or-self::title
26 return
```

Figure 4. BaseX search command (first lines) for *maiden* as part of combinations

Again, we do not explain all the technical conventions and details in lines 28 to 63. Suffice it to emphasise that our search formula here mainly consists of an <*if … then … else*>-function. The additional operator NOT in line 32 and the many tiered cross-references to previously-defined names (marked by the $ sign), add up to the whole search command appearing obscure for the uninitiated observer. However, in combination with the preceding algorithmic definitions (cf. Figure 4), the <*if … then … else*>-function is simply based on first-order logic (*for all … if … then*), introduced in its essentials by Frege (1879). The Innsbruck team managed to compose the query commands needed for all search criteria applied in *EDD Online*. While this was, to a high extent, a brainstorming and memory challenge, it would not have been possible without the help of BaseX, which mainly consisted in the software's permanent control of our programming steps.

3.2.3 Presentation of query result (c)

The output of the BaseX query command for *maiden*, popping up as shown in the (c)-box of Figure 3 once the query has been accepted by the system as valid, looks as follows (we have again selected the passage concerning *maiden-pasture*). Figure 6 shows that the query-command displayed in Figures 4 and 5 has filtered out exactly those details that are relevant for *maiden-pasture qua combination*: apart from details concerning the headword (such as

```
<result>
{
  $entry/(@n|prev|@select),
  $orth,$pos,
  if (not( $dialectExist))
  then
  (
      <compound>
      {
        <dialectFilter>{$dialect3}</dialectFilter>,
        $span,$combTerm,
        <labelFilter>{$label}</labelFilter>,
        <etymFilter>{$etym}</etymFilter>,
        <yearFilter>{$years}</yearFilter>,
        <timespanFilter>{<prev>{$prev}</prev>,<select>{$select}</select>}</timespanFilter>,
        <sourceFilter>{$title[term|country|district|region]}</sourceFilter>,
        $compPos
      }</compound>
   )
  else
   (
      <compound>
      {
      $entry/(@prev|@select),
      $span,$combTerm,$spannext,
      <dialectFilter>{$dialect}</dialectFilter>,
      <labelFilter>{$label}</labelFilter>,
      <etymFilter>{$etym}</etymFilter>,
      <yearFilter>{$years}</yearFilter>,
      <timespanFilter>{<prev>{$prev}</prev>,<select>{$select}</select>}</timespanFilter>,
      <sourceFilter>{$title[term|country|district|region]}</sourceFilter>,
      $compPos
      }</compound>)
     }

    </result>
```

Figure 5. Definition of output in a BaseX query for *maiden* as part of *combinations*

parts of speech), the *numerus currens* (9) of the combination, the combination itself, the dialect concerned (w.Yks.2), and all the filters that could be of interest for users in the context of a query for combinations, such as the *usage label* and *etymology* filter. As we did not activate any of these filters in the case of our sample, they are seen as being empty in Figure 6, with the exception of the *time spans* filter and the *source* filter. The *time spans* filter, by default, provides the earliest and the latest date of the sources mentioned in the entry concerned. So, when a user switches on the *time spans* filter, asking for a specific year or span of years, the computer 'knows' if and how these years overlap with those given in the entry.

The other filter with a value is the *source* filter. Given that *w.Yks.2* is a piece of information on both the source abbreviated by this string and the county dialect concerned (w.Yks.), it has to be grasped twice: as 'district' (= county[16]) and as 'source', sub-group 'title' (which stands for printed texts). This is the reason the *source* filter in Figure 6 has likewise been given a value.

4. The surface of human-machine interaction

On the basis of the BaseX result shown in Figure 6, other programs transfer the details into a surface convenient for human users. This is the task of web design. The attribution of scripts, format, colours, icons, boxes, run-down menus, and so forth, was created by one of our two programmers, Martin Köll, mostly in Java and Apache NetBeans. The output of his work is visualised in Figure 7.

```
<result n="E31723" select="1897">
  <orth n="W00003" facs="t=2600,b=2642,l=1609,r=1829" rend="bold">MAIDEN<pc n="W00004" ana="pc_prae" rend="bold">,</pc>
  </orth>
  <pos n="W00005" facs="t=2602,b=2646,l=1843,r=1907" rend="italic" style="abbr">sb.<pc n="W00006" ana="pc_prae">,</pc>
  </pos>
  <pos n="W00007" facs="t=2603,b=2648,l=1931,r=2003" rend="italic" style="abbr">adj.<surplus n="W00008" facs="t=2605,b=2640,l=2029,r=2106" ana="s_prae">and</surplus>
  </pos>
  <pos n="W00009" facs="t=2615,b=2641,l=2129,r=2168" rend="italic" style="abbr">v.</pos>
  <comb prev="1548" select="1897">
    <span n="W00152" facs="t=3116,b=3160,l=2656,r=2709" type="NUM">(9)</span>
    <term n="W00154" facs="t=3151,b=3193,l=1563,r=1741" rend="bold">{{Maiden}}-pasture</term>
    <span n="W00377" facs="t=319,b=353,l=1528,r=1577" rendition="startSource" type="NUM">(9)</span>
    <dialectFilter>
      <district n="W00379" facs="t=348,b=378,l=323,r=477" type="PREC">w.Yks.2</district>
    </dialectFilter>
    <labelFilter/>
    <etymFilter/>
    <yearFilter/>
    <timespanFilter>
      <prev>1548</prev>
      <select>1897</select>
    </timespanFilter>
    <sourceFilter>
      <title type="BIBLIO">
        <district n="W00379" facs="t=348,b=378,l=323,r=477" type="PREC">w.Yks.2</district>
      </title>
    </sourceFilter>
  </comb>
</result>
```

Figure 6: Extract of result window after the search of Figures 4 and 5

Figure 7 shows the selection of *maiden pasture* as number (9) of the listed combinations and the dialect area *Yks.2*, involved in *w.Yks.2*, correctly attributed to *Maiden-pasture*. Moreover, the figure shows, on top, the five open filters, that is, those that are not marked in red, from *dialect areas* to *time spans*. Philologists may find the visuality of Figure 7 a relief – after the challenging intricacies of the software notation in Figures 4 to 6. The design of our interface surface, the last step in our work, was partly a common-and-garden programming task, performed for any hotel and business website. This work needed a programmer's expertise in *Java*, *NetBeans*, and some other programming languages (for example, *Paint*, among other programs, for creating maps of dialect distribution).

However, in addition to programming routine work, Figure 7 may also be used to demonstrate the availability of a special 'hidden layer': the option of the *last-result* button (right of the search window). When this button is activated, our software 'knows' what entries loom behind the list of combinations in the retrieval window and allows for raising new questions within the framework of these entries. In such a 'nested' query all the restrictions on the eight parameters (left) and the filters (right), marked by red framing and imposed in the first query, are lifted so that a new grouping of parameters and filters could be arranged (within the framework of the previously selected entries). The hidden layers' intelligence consists in applying background results of a first query in a second one, and in switching off the previous blockage of a number of parameters and filters. There can also be further nested queries, so that we can say that the system, for several rounds, 'keeps in mind' data that the retrieval list does not expose.

An applied example of the usefulness of nested queries would be the retrieval of dialect forms that show 'h-dropping.' The user, in a first query round, will find 4,144 headwords beginning with an/h/; ensuing *last-result* queries for variants with an initial vowel (a, e, i, o, u) would elicit hundreds of forms with 'dropped' aches.[17]

In conclusion of this section, we hope to have shown that the software's extensive capability of searching for parameters and filters in correlation and combination with each other (as illustrated in Figure 7 for an intentionally simple example) testifies to the *EDD Online* software being able to 'learn', store and retrieve data according to the users' needs. Admittedly, the *EDD* is a small world compared to the global world of AI proper, but, essentially, they can be compared. They are both based on hard human work. In Big Data it is the work of masses of programmers who create the algorithms for evaluating the feedback of millions of online users. For *EDD Online*, the tasks can be described much more specifically: segmentation and classification of the multi-dimensional dictionary text (so far, classical structuralist work); then description of implicit hierarchies in the text (in analogy to sentence analysis in the way of Transformational-Generative Grammar); finally,

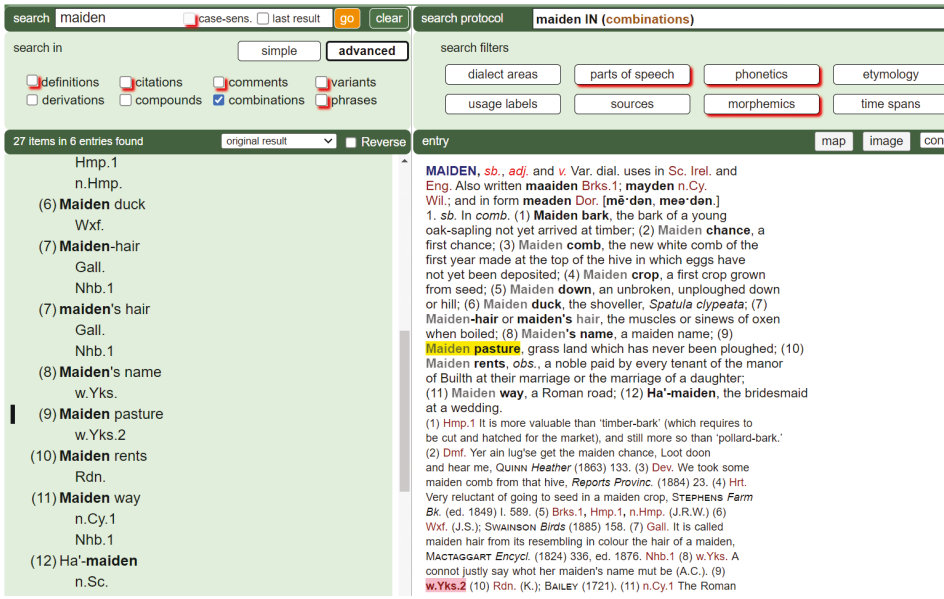

Figure 7. Interface of *EDD Online* after query for *maiden* as part of a *combination*

the translation of segments and hierarchies into 'strings' and mathematical rules, functions and expressions or signs. The *let*-rules of Figure 4, for example, are, in mathematical terms, equations: line 2, to select the one nearby, postulates that the tag *comb* be defined by, or should equal, the hierarchical expression that follows – as when we re-define the number 6 by 3x2. The *<if (not)… then … else>*-expression in Figure 5 is based on logical rules of induction: a rule and the output data are triggered by input data.[18] Finally, Figures 4 to 6 abound in technical expressions (such as *ancestor* and *descendant*) and IT-signs, such as the slash/, the vertical stroke | and brackets [] or parentheses () – they all have a specific meaning and mostly mark the role of the item concerned in the hierarchical structure of the text; like in mathematics: (a+b)*c is different from a+b*c.

We as philologists would have preferred to relegate explanations of the technical type on the last few pages to IT experts. In our work, however, this proved not to be feasible. Programmers usually will not know and tend to be less interested in the hierarchical structures of dictionary entries. In our Innsbruck team, the philologists Andrea Krapf and Manfred Markus thus simply had to take over some of the programming tasks. IT-specialists may find the explanations in this section of our paper superficial and incomplete, just as philologically trained readers may consider them to be too technical. However, in line with the interdisciplinary tasks, the team members of the Innsbruck project tried to bridge the gap between the two camps by crossing methodological borders to the best of their abilities.

By contrast, in the big world of commercial AI, there is a clear division of labour. ChatGPT, for example, is coached to be 'intelligent' by thousands of 'trainers,' who feed the servers, that is, the firms running the servers, with 'correct' data – correct in any sense of the word, including political and moral 'correctness.'[19] The computer 'learns' or 'understands' this correctness of data by induction. With its superhuman memory, it stores and thus is trained to ignore data, including faces, texts, and thoughts, that have been put on the 'index' by the coaches, who, we understand, do not even know what their (arguably badly paid) work is needed for.

5. Comprehensive dialectology with artificially intelligent *EDD Online 4.0*

The value of a comprehensive approach in dialectology as suggested here can hardly be overestimated, with 'comprehensive' referring to a range of accomplishments, e.g. (a) that a great number of dialect areas can be compared with one another; (b) that different dialect features can jointly be examined; (c) that all kinds of statistics on dialect matters can easily be provided so that an objective picture of dialect areas can be drawn; (d) that dialect features are retrievable not only across space and time but according to their full linguistic (and partly cultural) characteristics; (e) and finally that the distribution of features in space can be quantified on a normalising basis and dynamically visualised by different types of maps. *EDD Online 4.0* can certainly claim to be comprehensive along these lines.

By contrast, traditional dialectology is characterised by its method of 'manual' analysis, for example of a word or lexical variant, of a way of pronunciation or feature of accent, or of whatever else is traced to represent a certain dialect area. A slightly more challenging approach has been analysing multiple features in succession. However, this kind of description neither covers all features of a given dialect area nor does it measure if and to what extent the selected features are area-specific ones. What we need is more synthesis of observation and quantification of the findings. Quoting several samples of an assumed feature, one sample after the other, is no remedy against the limits of the traditional manual method.[20]

However, a comprehensive analysis is more easily claimed than done. Dialect has many facets, and if they are all considered at the same time, results are bound to be chaotic and confusing. This being said, in order to avoid confusion, we need an order of the features and some rules to reduce the – theoretically – countless number of combinations of features.

5.1 Avoiding the combinatorial trap

EDD Online, for its part, provides ten search parameters and eight filters which include many sub-filters and sub-sub-filters. The search parameters are the two 'simple' ones, *headword* and *full text*, and the eight 'advanced' parameters, which we have called *definitions, citations, comments, variants, derivations, compounds, combinations,* and *phrases*. The eight filters mostly refer to grammatical and usage criteria of the parameters just listed: *dialect areas, parts of speech, phonetics, etymology, usage labels, sources, morphemes,* and *time spans*. The subdivision of these filters varies. *Dialect areas*, for example, has the three sub-filters *country, region,* and *nation*, with overall 130 counties, 42 regions, and 14 nations distinguished within these sub-filters. The filter *usage labels* contains eight subtypes, from *frequency* to *syntax*, with well over a hundred sub-sub-filters included in them. In a similar way, the other filters subdivide into large numbers of search criteria. The filter *sources* is particularly concerned by sub-classification, owing to Joseph Wright's extensive use of source types, both printed and unprinted, and of individual sources.

Readers of this paper need not be fully familiarised with the contents of all the search parameters and filters of *EDD Online*. The examples mentioned may suffice in demonstrating that the *EDD* provides extremely multi-dimensional information and that the possibility of combining many (though not all) of the search criteria with each other ends up in what is known in complexity theory as 'combinatorial explosion' (Raggett & Bains 1992: 44f.).[21] There are several ways of avoiding the 'combinatorial trap'. One consists in reducing the number of items by cover terms. In *EDD Online*, we have, for example, reduced the number of references to Yorkshire by subsuming, in the way of fuzzy logics, all references to the whole as well as parts of Yorkshire, also including towns in Yorkshire. Figure 8 illustrates this all-inclusive definition of the counties.

The same principle is applied to the other filters. In the filter *usage labels*, for example, the sub filter *semantics* addresses the keyword *children* by also retrieving hyponyms, such as *boy*, and morphological or syntactic variants, such as the plural *girls* and the phrase *female child*.

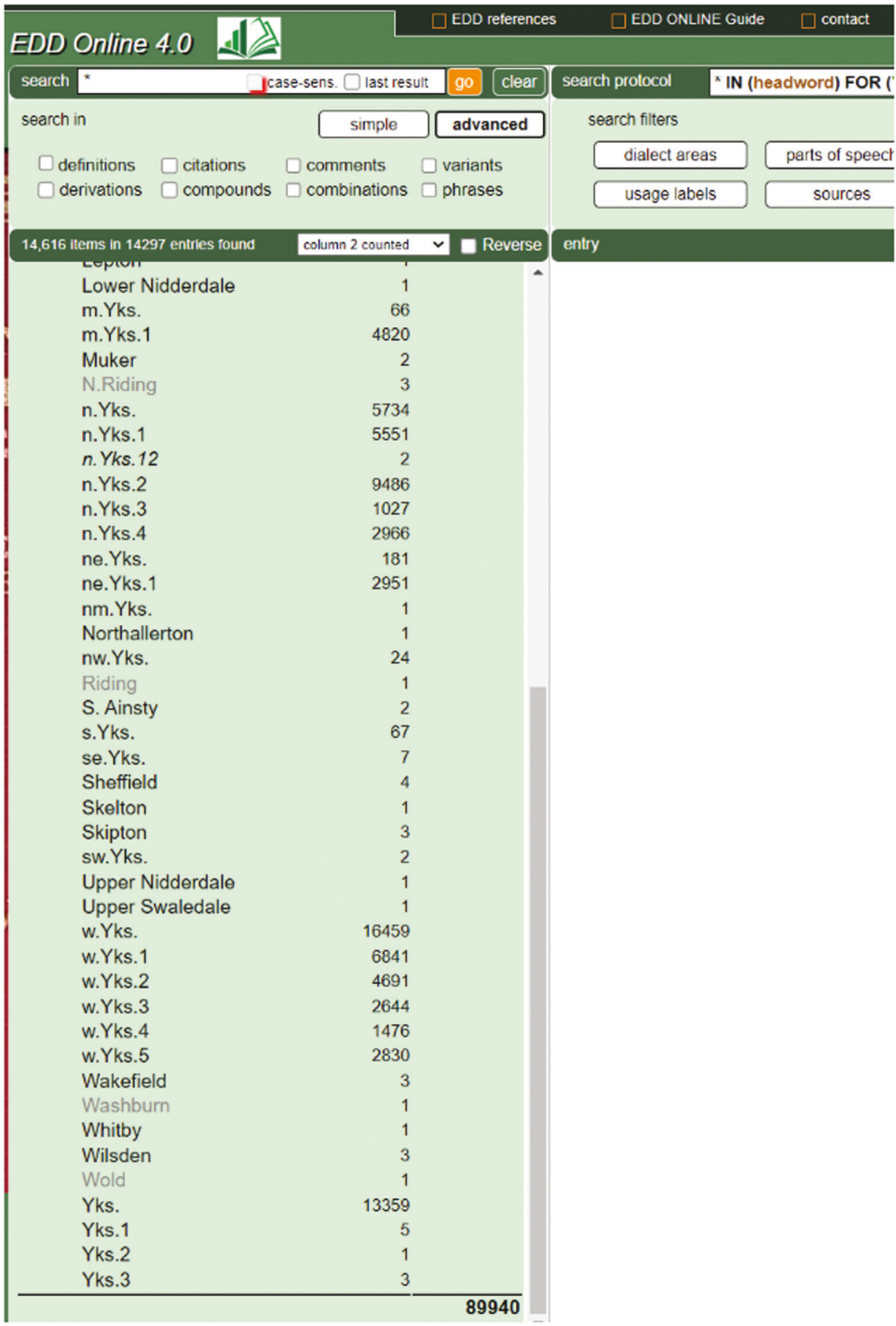

Figure 8. Result of a search for headwords attested to Yorkshire (extract)

Figure 9 illustrates that, while most of the findings have been retrieved as one-word strings, there are others within phrasal contexts that were less easy to find. For these, special query rules had to be designed. The distinction between the type of items called *other* and *various/total* will be discussed below.

The compilation of certain 'tokens' subsumed under a 'type' in order to reduce the number of query processes has also been practised in the filter *etymology*, to add a further example. Searching for headwords/entries with German cognates provides a retrieval list of 297 references to etymons in standard German or in German dialects.

Another method of reducing the number of possible combinations to avoid the 'combinatorial trap' is to block those that do not make sense. When, in a 'simple' query routine, we switch to *full text*, the software will 'understand' that this type of search implies a plain scanning of strings, one after the other through the whole dictionary, without considering any structural aspects. Accordingly, all the filters are blocked, see Figure 10.

The advanced parameters are a bit more challenging. A search for strings in *definitions* excludes information on any of the other parameters and, on the side of the filters, blocks four of them (Figure 11). The example of Figure 11 shows that the parameters and filters that are excluded from combined searches for defining terms are formal or structural ones. Our software 'understands' that such aspects have to be kept apart from semantic query terms.

This principle of sensible reduction of query criteria is, as the example shows, far from being merely mechanical. In Figure 11, we have combined a search for the defining string 'house' with all English counties as a filter criterion. In the result list on the left, we see that *house* is used in the first definition of the entry AGATE. The *numerus currens* (1) for *Bead-house* in the entry window (on the right) is important because the abbreviations of English counties then attributed to this number are those that are provided in the subsequent paragraph.

The correct correlation of pieces of information by the same index number was not at all simple. There are counting lists in the *EDD* on six different levels: Roman numbers, Arabic numbers, Arabic numbers in parentheses, capital letters, small letters in parentheses, and occasionally Greek letters in parentheses. Figure 12 shows one of the problems that our programming had to solve under these conditions.

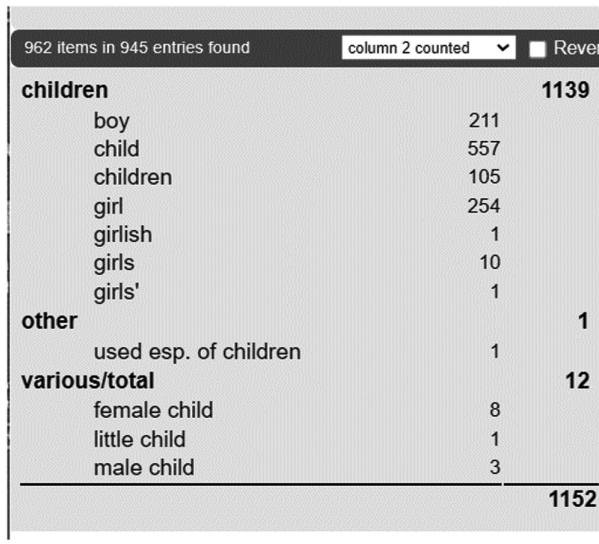

Figure 9. Search for *children* as a semantic feature (semanteme)

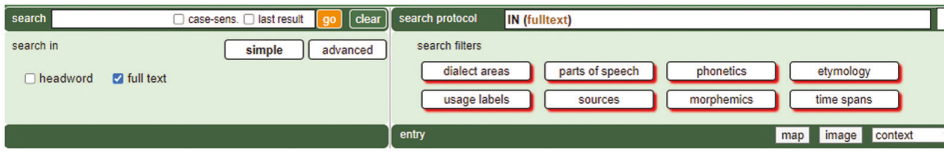

Figure 10. Blocking mechanism in a search for strings in *full text*

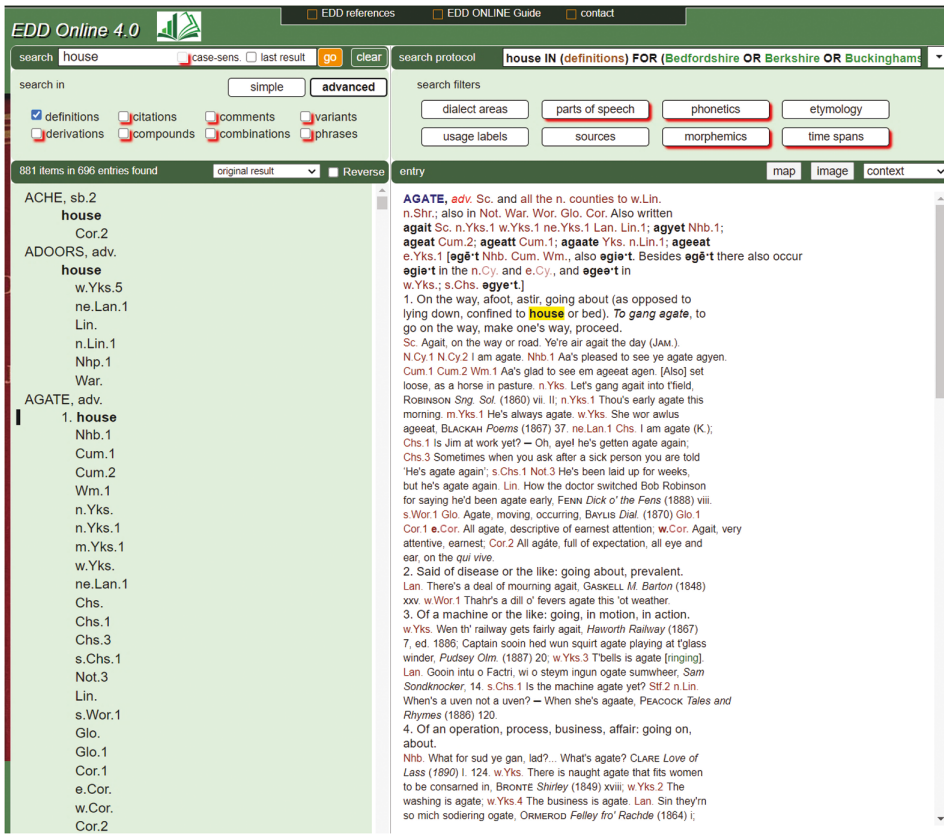

Figure 11. Search for defining string *house*, with four filters blocked

The query for compounds with the string *house* in them has provided 163 items (as shown above the retrieval list), one of which is *bead-house*. The entry-window on the right has highlighted the correct English counties though the index number in parentheses is 'disturbed' by the inferior counting in letters. Our software had to 'learn' that the counting number '(1, a)' is as good as '(1)' and that its range includes both (a) and (b), because these contain information on meanings both covered by the compound *bead-house*. In other words, the software has to 'know' the hierarchical relationships between the textual elements in order to correlate them correctly. As generally shown in the last section, this 'knowledge' is provided by the hierarchically structured XML-tags of the linear dictionary text and the derived hierarchical structure of the TEI-version.

Given the many search parameters and filters offered in *EDD Online*, there would be numerous other examples that would show the 'intelligence' needed by our software in order to retrieve sensible information. In the following, we will select two further examples.

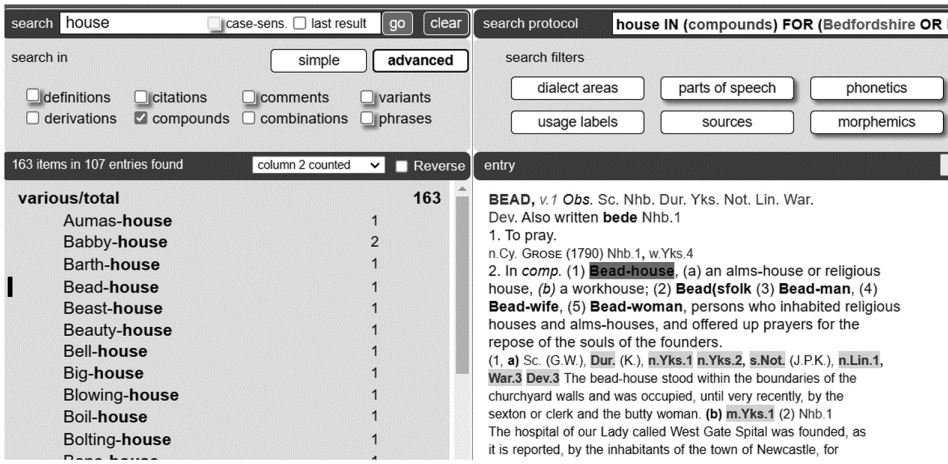

Figure 12. Complex correlation of a compound with the dialects of its distribution by means of counting numbers and letters

5.2 A focus on usage labels

It is common practise in Present-Day English dictionaries to characterise headwords by labels. While some labels are often used, such as *C/U* for *countable/uncountable* nouns and *tr/itr* for *transitive/intransitive* verbs, there seem to be no rules for what exactly should be labelled and by what abbreviations. Wright did not have any style sheet for his labelling practise either, so that, given the large number of very specific labels, the philological members of the Innsbruck project had to classify them: *frequency, reliability, semantics, pragmatics, phonology, prosody, morphology,* and *syntax*. *Pragmatics* proved to be a particularly fertile ground – as Figure 13 suggests.

Our software finds more than a thousand items and more than 1,100 passages with reference to *children* as a pragmatic piece of information. So, how would it 'know' that the strings or phrases listed under *children* in this case have pragmatic implications, rather than semantic ones, as in Figure 9? The answer is that the strings were tagged accordingly. This was semi-automatic work, first relying on certain phrases flanking the keywords *boy, girl, child** etc. in the text – e.g., phrases such as *used by* and *applied by* – and then checking the attributed tags individually. The borderline between semantic and pragmatic information is and was subtle and difficult to draw: The definition of a term that is 'used *or* applied **by** *or* **for** children' is of pragmatic relevance, whereas an explanation saying that a term is 'used *or* applied **to** children' has semantic implications. Such a context-conditioned interpretation of keywords or key-phrases, such as *children's game*, could only be carried out up to a point. The words and phrases that we did grasp appear in Figure 13 under the cover term *children*. In two other cases there was an overlapping with other hyperonymically pragmatic keywords – in our case with *threat* and *warning* –, so that the retrieved items are primarily subsumed under these hyperonyms though also listed in the result list for *children*. Finally, there are some further cases, of phrases which we failed to identify individually, but which contain some string that suggests pragmatic children-related relevance of the phrase concerned. We tentatively added such cases under the heading *various/total* or, rather, let the program add them, risking to provide occasionally questionable or invalid findings. Here it is up to users to check whether the computer's mechanical sorting according to simple keyword strings is always valid.

This human-machine division of labour has generally been aimed at in *EDD Online 4.0*. However, the filter *usage labels*, with its extreme ramification, was in particular need of a

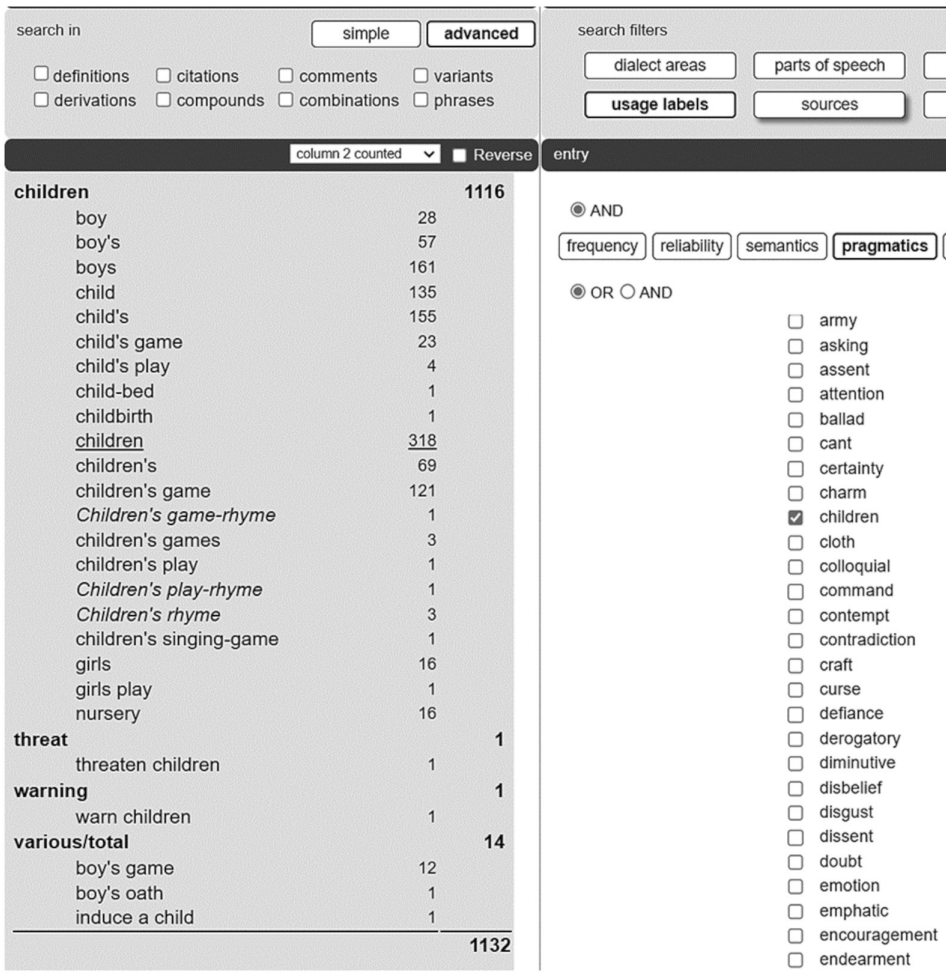

Figure 13. Search for the label *children* as a pragmatic feature

retrieval procedure that leaves space for merely probable or open cases. This is why we had the headings 'other' and 'various/total' in the list of semantic retrievals in Figure 13 as well.

Another wide field of data, apart from *usage labels*, is provided in the filter *sources*, owing to Wright's use of thousands of printed and unprinted documents and also owing to his, or his team's, variable way of referring to them. Here again clear references could be grasped mechanically by our software, but, in repeated phases, the conundrum of information given by abbreviations had to be averted in such a way that the data were graspable by the machine. Again, many cases had to be left open after careful manual investigation.

6. Creating virtual maps – a specific capability of *EDD Online*

Google Maps is a widely used and highly dynamic web mapping platform: you type in an address or the name of a small village, and Google shows you on a map where it is. The size of the map is flexible and has different tiers or layers: you can switch to satellite mode, activate positions of hotels, filling stations, and so forth. In fact, the concept of multiple map layers stored in so-called Geographic Information Systems (GIS) is a key feature of

most modern maps (Werner 2022). In our case of *EDD Online*, the layers are not streets or hotels, but, first, the three options of a political, geographical, and hybrid visualization. Second, the system offers the three types of 'zoning': counties, regions, and nations, in addition to the two base maps, UK and the world, which are automatically activated. Unlike Google, we do not have a satellite mode, but we have implemented a zoom-option and, above all, an areal choropleth visualisation, which automatically transfers graded quantities into colours and, in the 'relative' mode, has three levels of normalisation on offer.

GIS technology is closely connected with AI. In an article on AI GIS in the *Journal of Geo-Information Science* (Guanfu 2020), the editor of the journal explains that three geospatial machine learning operators can be distinguished (Figure 14).

While we do not claim to fully comprehend all aspects of Guanfu's paper, we understand that spatial clustering and classification are part of what *EDD Online* achieves, and that regression is involved when a map created by a query for dialect areas is modified by additional filters, i.e., layers, of selection. In fact, each of the eight *EDD Online* filters and of the hundreds of sub-filters can be interpreted as a regressive layer. A search for all dialect areas available in *EDD Online* but restricted by some other filters (e.g., *sources*), can be seen as a case of 'regression.' Also, the restriction is 'weighted' in that sources (to stay with them) are excluded if they have a frequency rate of < 200. Moreover, the ONLY operator, which is available in our interface in addition to OR and AND, is able to reduce the findings of a first-round query (let us say, a query for Yorkshire) if (and only if) no other dialect area co-occurs in the given entries.

This concept of multiple map layers is, thus, key for both, GIS and *EDD Online*. As says Werner 2022 about GIS: 'Each map represents a different group of geo-referenced data points that are superimposed one on top of another to create a composite map image.' This generation of composite *ad hoc* maps in colour when and in the way needed by the user, is, to our knowledge, a rare novelty in dialectological atlas work. In the face of a plethora of language atlases created with the help of computer software and recently published or announced, the following advantages of *EDD Online* should be pointed out:

[1] The *EDD Online* maps are not based on maps of previously published atlases, as in the case of *The Computer Developed Linguistic Atlas of English*, based on Viereck and Ramisch 1997 (see Embleton and Wheeler 1997) the *Finnish Online Dialect Atlas* [Embleton and Wheeler 2000]), but are original maps created dynamically and on the basis of multiple hidden layers of selection.
[2] *EDD Online* is presently on offer (unlike RODA, the *Romanian Online Dialect Atlas* [Embleton, Uritescu, and Wheeler 2007]).
[3] *EDD Online* maps are quantitative choropleth maps and thus easily interpretable by users. Quantitative areal maps are "the most fertile area for the development of dialect maps […] characterized by the application of statistics and modern GIS

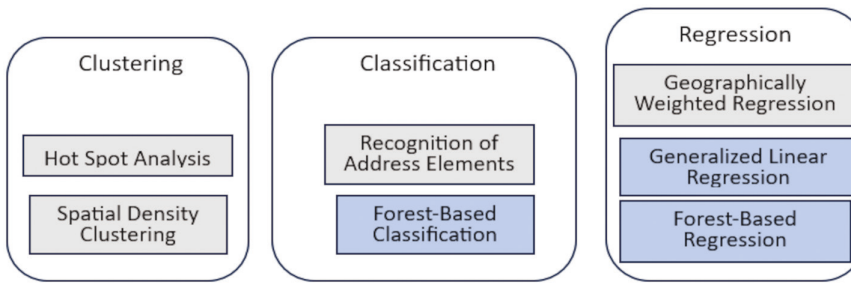

Figure 14. Role of clustering, classification, and regression in AI GIS

technology" (Rabanus 2018: 356) Choropleth (i.e., multi-colour) maps are an excellent means of illustrating the geographical distribution of dialect features according to subtle quantification. If, for example, we take the imperative form of verbs as a syntactic feature, searching for * (headwords) in combination with all counties and regions of the UK (incl. Ireland), the map that can be generated *ad hoc* to illustrate the distribution is most revealing (see Figure 15).

Which brings us to the applied part of this section. Figure 15 shows, in its three windows, first, at the right top [= (a)], a documentation of the many counties and regions involved in our search, as well as the Boolean operators OR and AND (OR for the combination of the dialect areas, AND for the additional combination with the feature *imperative* (at the end of the protocol).[22] Second, we see, in the left half, the quantified list of types/headings and tokens concerned by our query [= (b)]. This quantification allows for the *ad hoc* production of the map in the third window of Figure 15 [= (c)]. The map discloses at a glimpse that while the colour blue is dominant in England, there are remarkably deviant colours in Ireland, Scotland, and Wales. As our query has included regions in addition to counties, the map not only has coloured counties, but also delineated regions, for the sake of clarity marked by a coloured dot. The colour of the dot stands for grades of frequency, just like the colours of the counties and in line with the legend visible in the top left corner of the map.

According to the map the particularly deviant counties and regions are Cromarty and the Highlands in Scotland; Wexford, Queen's County, and Waterford in Ireland; and South Wales in Wales. The Isle of Man and Oxfordshire are also deviant, but less so. There may be deeply cultural reason for these results, but the discussion of reasons for

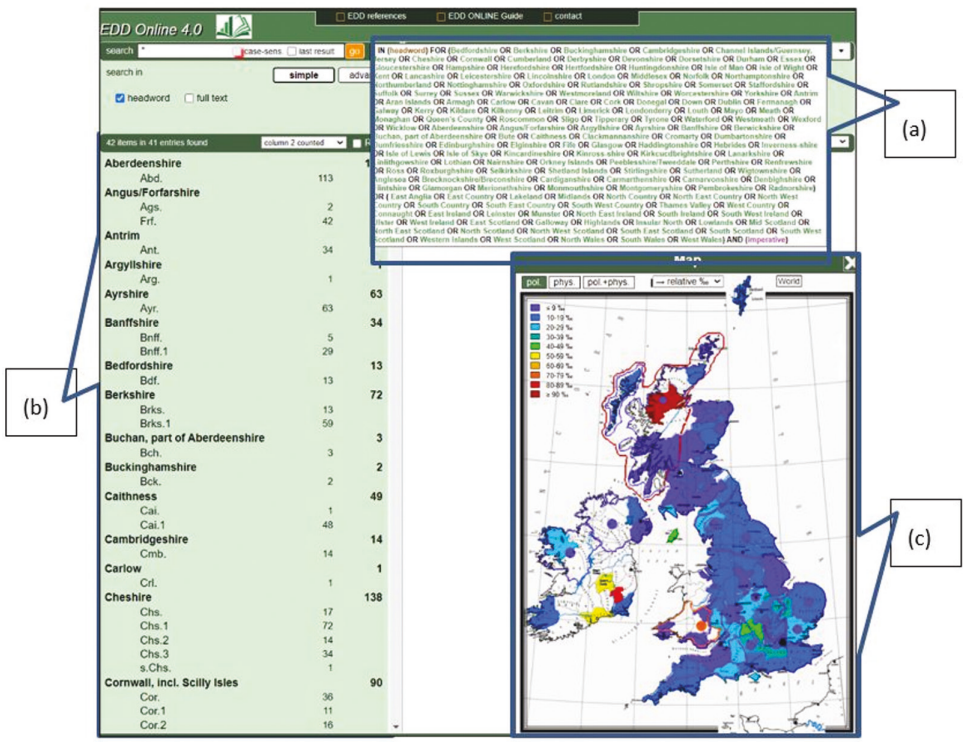

Figure 15. Map showing the distribution of imperative constructions in the UK

retrieval results goes well beyond the aim of this paper. Instead, the 'relative' quantification, selected for the map of Figure 15, needs some explanation. Actually, there are four options of quantification: an 'absolute' one, and three 'relative' ones. 'Absolute' means that the frequency number for an area, say the number 63 for Ayrshire, is only related to the sum total of dialect references, which is always given at the end of the retrieval list (and not visible in Figure 15). This number, which is 6,034 in the case of the query conducted in Figure 15, is of little interest since it depends on the random selection of dialect areas in a single query.

By contrast, the more interesting 'relative' figures are, behind the scene, related to the number of overall occurrences of references to each of the given areas; for Ayrshire this number would be 7,951. When users move their cursor over a coloured area of their *ad hoc* map, a smart tag mechanism pops up, informing them about the number just mentioned, and automatically figuring out the percentage or *per mille* or *per ten thousand* weight of the frequency number for the given area. In Figure 16, we have moved the cursor to *Ayrshire*, thus triggering the information that the number 63 (of Ayrshire references related to the syntactic marker *imperative*) is 'worth' 7.9 *per mille* in relation to the total of references to Ayrshire. This process of normalisation is triggered for all filters whenever the attribution of dialect areas makes sense.

Just as with usage labels, the computer accomplishes many tasks for us, correlating dozens of dialect areas with over 100,000 words, with specific filters, and with total numbers of dialect references looming in the background. Yet, users have to decide for themselves which of the three 'relative' modes is best to create a sensible profile of regional distribution – if the colours are all more or less the same, a change of the statistical *quorum* may create distinctive colours. Of course, when we search for very frequent dialect features, the percentage mode (rather than per mille or per 10,000) is the best measure to illustrate differences.

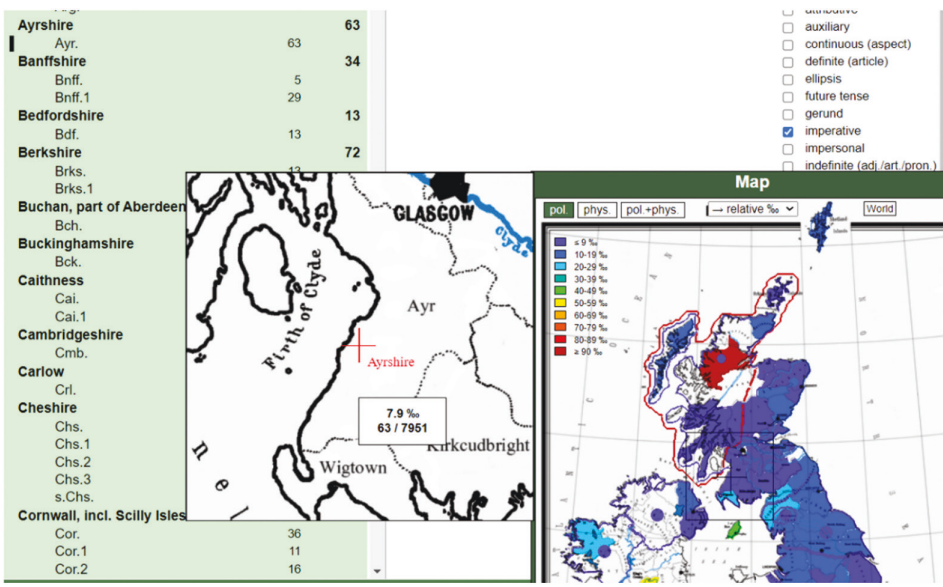

Figure 16: Smart tag information on 'relative' *per mille* frequency

Striking evidence of this is provided by a query for all headwords in the *EDD* that are (also) verbs. This criterion is triggered by the option *verb* in the filter *parts of speech*. Figure 17 suggests that England is more 'verbal' in dialect than the other countries of the UK.

One could deepen this kind of analysis. If England was really more 'verbal' in dialect lexis, particularly compared to Scotland, would Scotland then prove, in another query, to be more 'nominal'? In the face of such general hypothetical assumptions, one should, however, see the risk of a 'biographical fallacy.' It could well be that Wright and his team, with their English background, were simply more interested in going into depth on English words, whereas for Scotland, Ireland, and Wales they were tendentially keen on more basic lexis, focusing on nouns. Food for future thought.

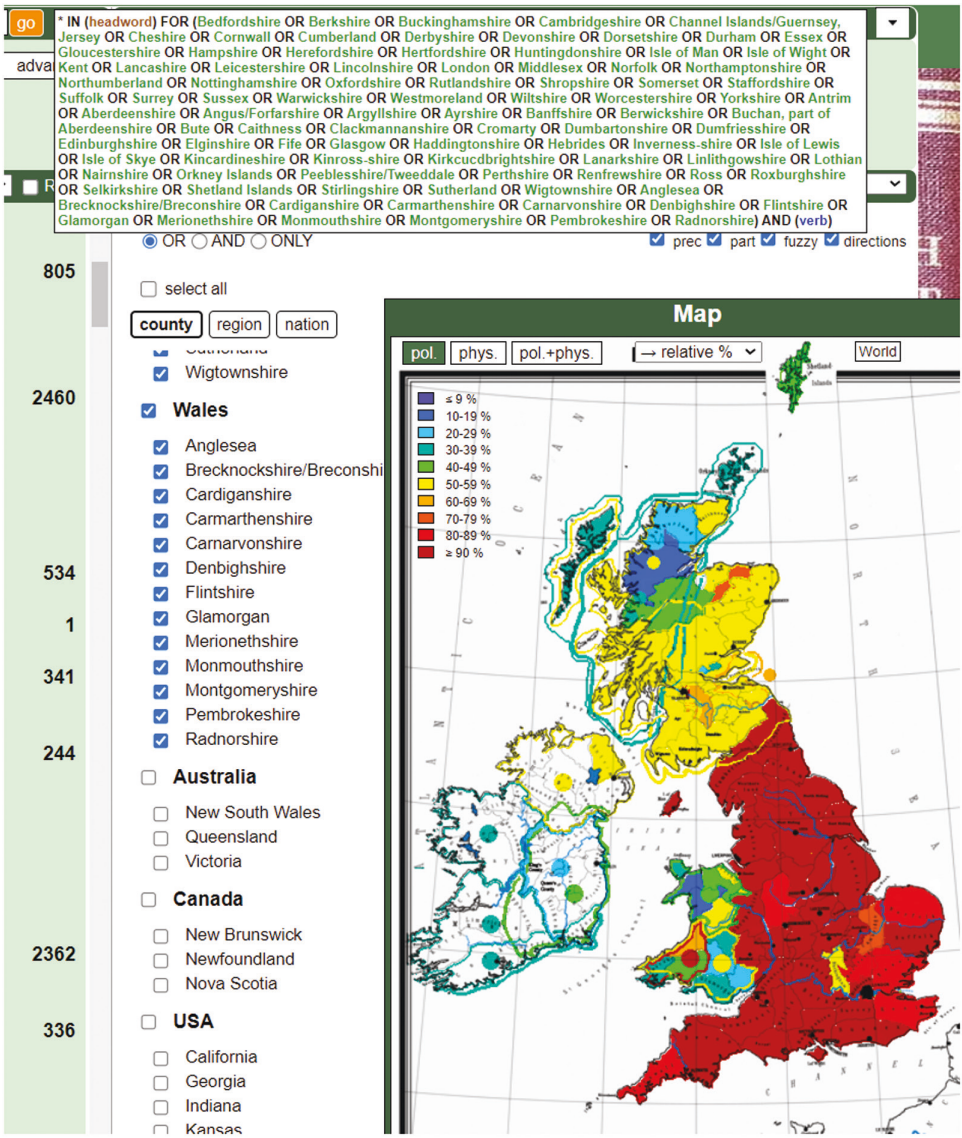

Figure 17. Result of a search for verbal headwords with percentage mode of map presentation

7. Summary and conclusion

Regional dialect varieties, compared to standards, are the more natural versions of languages. They are, however, a most complex matter in that they are multifaceted, permanently change and yet are often surprisingly conservative. Their features come and go, and in their historicity, they are hard to trace. Their inaccessibility is increased by the fact that, since the late 19th century, they are superimposed by socially based varieties. As a result of these factors, regional dialects, while often described by amateurs, have generally been studied eclectically, that is non-comprehensively. Their complexity has disallowed a comprehensive approach.

With Wright's achievement of the *EDD* and the interface of *EDD Online*, the situation has radically changed. Created by linguists in cooperation with programmers, the Innsbruck computer platform can be claimed to be an 'intelligent' toolkit that allows users to raise questions unanswerable by a human scholar by way of any traditional approach – questions on thousands of words and sources and hundreds of dialect areas, usage labels and other dialect criteria at a time. Not all these tools are AI-based, but we claim that some of them are 'Artificial Intelligence' at its best – created by humans in a fertile and transparent cooperation with the 'machine' and also functioning in this cooperative way. However, like ChatGPT, which has lately often been presented as being equipped with an allegedly autonomous brain, *EDD Online*, with all its superhuman possibilities, clearly has its limits. *EDD Online* (4.0) is a fascinating vehicle, using AI of the traditional type and scratching the surface of its recent achievements; but it still needs humans (the users) at its steering wheel.

In the present paper, we started by reflecting the term 'Artificial Intelligence' from a general point of view beyond that of Computer Technology. We then, in an interdisciplinary description, shed light on the software used for *EDD Online*, to the extent that philologists could learn and understand IT 'knots.' Subsequently, we described, in due brevity, the complexity of the possible search routines of *EDD Online*, providing an overview of the ten parameters and the eight filters of our interface. The examples of analysis then used represent a fairly random selection. We discussed the complexity of the *usage label* markers and of the *ad hoc* maps created by implemented algorithms of quantification, maps that allow for a comparison to modern GIS methods. Instead of *usage labels*, we could as well have selected *sources* as a filter to illustrate the necessity of human-machine interaction.

The *usage labels* and their eight sub-filters were shown to be a most important dimension of information in the *EDD*, apart from dialectal areas. Focusing on pragmatic labels and the keyword *children* as an example, we demonstrated the share of previous manual tagging and the division line between what the computer can do and what is left to the users' judgement – the computer 'relies on' frequent strings, whereas we as the creators of the system, and the users as its consumers, take care of the exceptional, rare, and complex cases.

Our view on the role of *ad hoc* maps in *EDD Online* was based on a representative search for imperatives and, in a short second step, verbal constructions related to all counties and all regions in the UK. The strength of the computer was shown to consist in its ability to grasp dozens or even hundreds of dialectal areas at a time, to arrange them, after their retrieval, according to their normalised names by attributing all the abbreviations to them, and in its ability to map them according to explicit or implicit quantification. Implicit quantification came to bear when our software related quantities of retrievals to the total frequencies of occurrence of each of the areas concerned. The computer 'knew' these total frequencies because we had been storing them, but the different types of mapping and the filtering of area frequencies by whichever other filters or layers proved feasible thanks to our smart software.

In conclusion, the software of *EDD Online* is 'intelligent' in 'knowing' and completing tasks quickly, comprehensively and in one go. A glimpse at other computerised dictionaries, for example, the *Oxford English Dictionary (OED)* and *Deutsches Wörterbuch* (Grimm

and Grimm 1854–1960) reveal the persistent methodological traditionalism. In the case of *Deutsches Wörterbuch*, the dominance of the units of entries and their headwords, at the cost of linguistic parameters, is obvious.[23] For the *OED*, Markus (2021b: 268) has shown its limits as regards dialectal lexis and, in particular, dialectal features.[24] Both dictionaries have a historical, that is, temporal rather than a spatial focus. Like in modern computerised learning dictionaries, dialects play a marginal role. In decidedly variety-based English dictionaries, such as DARE and eWAVE, the interactive possibilities of the users are coarse-grained and rather limited.[25]Again, headwords are in the centre of interest, at the cost of types of word formation (e.g., compounds). Linguistic features cannot be mapped in combination but only singularly. And both projects work with a limited number of icons standing for absolute frequencies, rather than with dynamic choropleth maps representing normalised quantities of occurrence. Finally, the *Survey of English Dialects* was made accessible in 2019 by the Salzburg 'dialectometrist' Hans Goebl and his team.[26] This approach, also offered for other European dialects than the English ones, is, from a philological point of view, an extremely claimant mathematical method of describing similarities of, and differences between, dialectal areas (cf. Markus 2018). Given the abstractness of Goebl's model as well as its focus up to now on the analysis of sounds, and the questionable role of repeated clustering, users may find it difficult or even impossible to correlate areas to concrete dialect features of lexis.[27] This model enables the recognition of overall similarities and differences across the accents of a given area, but owing to its abstractness, there seems to be no way to flexibly shift from hyper-complex synthesis to lexicographically relevant analysis, the least for English dialectal lexicography.[28]

To our knowledge, then, *EDD Online* is the only 'intelligent' dialect dictionary that exists. However, like with the self-driving car, the 'brain power' behind its behavioural competence is human, in the shape of programmers or 'controllers' of the system itself. There are differences of degree and not of kind to ChatGPT and other versions of AI. One of the differences we do concede and would in fact like to emphasize is that ChatGPT is a black box, non-transparent – unlike *EDD Online* – to practically all its users. However, people like to be illusioned by seemingly perfect simulation of human creativity, and are fascinated by enigmatic machines that can outperform humans in hitherto uniquely human skills. It is true, in many sections of life, machines may and will increasingly do better than human beings. But a tool, never mind how sophisticated, will remain a tool; it lacks the ability of being creative and, above all, of assuming responsibility. As humans, however, we **are** responsible and liable for what we create – arguably, like Victor Frankenstein was.

Notes

1 It is, in fact, not at all new and had its first peak in the 1970s and 80s (cf. e.g. Raggett and Bains 1992: VIII).

2 As Ing & Grossman (2023: 3) summarise in their preface, '[c]learly, the potential for robots and AI to improve the quality of life is enormous.' For further recent literature see their list of references (2023: 12-14). On the impact of ChatGPT on lexicography see De Schryver & Joffe (2023).

3 Otte, though a renowned AI expert, comes to a similar conclusion: 'AI will neither overtake us intellectually nor is it far from being a match for us in our natural environment.' (2021: 235; translation by the authors).

4 Cf. https://iblnews.org/chatgpt-is-high-tech-plagiarism-it-undermines-education-said-noam-chomsky/ (accessed 25 Oct 2023). The full interview is accessible here: https://www.youtube.com/watch?v=IgxzcOugvEI.

5 Cf. https://the-decoder.com/chatgpt-is-high-tech-plagiarism-for-lazy-learners-says-noam-chomsky/ (accessed 25 Oct 2023).

6 For an in-depth discussion of such questions, see e.g. Harari (2019: 48-50, 301-312), Russell (2019), and Simanowski (2020).

7 'Feed-forward' means that the flow of information between the layers of a network is unidirectional, flowing from the input nodes through the hidden nodes to the output nodes, without any loops or cycles (see Zell 1994: 73).

8 Cf. https://www.scientificamerican.com/article/ai-is-getting-powerful-but-can-researchers-make-it-principled/ (accessed 3 Nov 2023).
9 The URL-address of the latest version is eddonline4-proj.uibk.ac.at; its use is free of charge. It was supported by the Austrian Science Fund (PUD 24-G).
10 XQuery is the 'language' (the lexis and syntax) for handling XML texts; BaseX is the tool for filtering out data from these texts. For details on XQuery see Walmsley (2007).
11 *Combination* is the term used by Wright for a type of word formation whose elements are not as tightly joined as in the case of a *compound*. For details, see Markus (2021a: 90-91), and Onysko (2010: 143).
12 'Partly' because we used the XML browser and editor *Oxygen* to automatise our correction work as much as possible.
13 And is, thus, too long to be presented here. But see Appendix.
14 An *attribute*, unlike the (conventionalised) *tag*, is a freely definable code for characteristics of less frequent elements of a text; they are marked by @, as in @type='COMB' in the second line of Figure 4. Here an attribute was needed because no tag was available for *combinations* (as a type of word formation) in TEI.
15 For example, cf. Harold and Means (2004).
16 TEI does not tolerate the item 'county', so we had to replace it by 'district.'
17 The retrieval of *h*-dropping in *EDD Online* was discussed in Markus (2021a: 101-111).
18 Cf. the theoretical role of *induction* for AI in Otte (2021: 98-100).
19 In the German periodical *Der Spiegel* (29; 15.7.2023, pp. 58-61), a detailed article described 'the faces behind AI'), that is, the click-workers and content trainers, engaged by firms such as OpenAI, Amazon, and Google. Cf. also Gray and Suri (2019).
20 For a more detailed survey of recent English dialectology based on feature analysis, cf. Markus (2022).
21 Raggett and Bains used the adjective *combinational*, but according to recent online contributions to the issue, *combinatorial explosion* is now the technical term. It refers to the fact that the number of possible combinations of items extremely rapidly increases with the number of items.
22 We are aware that the dialect attribution often does not **directly** refer to the feature *imperative*, but only indirectly, sharing the same entry with the dialect references.
23 The query options are extremely limited. Users can search for full text, keywords (in general), word classes, strings from quotations and from sources (see https://woerterbuchnetz.de/?sigle=DWB#NaN), but linguistic criteria remain totally hidden in the extremely long articles.
24 The analysis is summed up as follows: 'OED3 comes with a great many options allowing specific perspectives, but the regional distribution of forms and meanings, including the reference to counties, is eclectically included, but not consistently retrievable.' (Markus 2021b: 268)
25 DARE is the *Dictionary of Regional American English* (1985-2013); eWave is *The Electronic World Atlas of Varieties of English* (Kortmann et al. 2020).
26 Cf. https://dialektkarten.ch/dmviewer/eng/index.de.html
27 The abstractness consists in the polygonic definition of dialect areas (beyond areas in nature), the fixation of a spatial reference point (in relation to which deviations are measured), the definition of an arithmetic means in deviation, and so forth; linguistic features are non-distinctively itemised as *Total Phonetics*, *Consonantism*, *Vocalism*, and *Vocabulary*.
28 Also cf. John Nerbonne as one of the key figures in dialectometry, who has published widely on this topic over the last years (e.g. Nerbonne 2015, Nerbonne et al. 2021).

References

A.Dictionaries and word lists

Dictionary of American Regional English [DARE]. 1985-2013. 6 vols. Cambridge, Massachusetts: Belknap Press of Harvard University Press. https://www.daredictionary.com/

Grimm, J., and W. Grimm. 1854–1960. *Deutsches Wörterbuch*. Available online from Kompetenzzentrum Trier. 2023. *Deutsches Wörterbuch von Jacob Grimm und Wilhelm Grimm auf CD-ROM und im Internet*. https://woerterbuchnetz.de/?sigle=DWB#0 (accessed 5 August 2023).

Markus, M. and team. 2023. *English Dialect Dictionary Online 4.0*. https://eddonline4-proj.uibk.ac.at (last accessed 20 August 2023).

Oxford English Dictionary [OED], version 3.0: https://www.oed.com/. Last accessed 2. August 2023.

Wright, Joseph. 1898-1905. *English Dialect Dictionary*. Oxford: Henry Frowde.

B.Other literature

Antoniadis, P. 2022. 'Hidden Layers in a Neural Network.' https://www.baeldung.com/cs/hidden-layers-neural-network (accessed 26 Oct 2023).

Bastian, M. 2023. 'ChatGPT is 'high-tech plagiarism' for lazy learners, says Noam Chomsky'. *The Decoder* Feb 13, 2023 (https://the-decoder.com/chatgpt-is-high-tech-plagiarism-for-lazy-learners-says-noam-chomsky/ (accessed 31 Oct 2923)

Carlbring, P., H. Hadjistavropoulos, A. Kleiboer, and G. Andersson. 2023. 'A new era in Internet interventions: The advent of Chat-GPT and AI-assisted therapist guidance.' *Internet Interventions* 32: 1-33.

CIRP Encyclopedia of Production Engineering. 2014. The International Academy for Production Engineering, L. Laperrière, and G. Reinhart, eds. Berlin, Heidelberg: Springer. DOI: https://link.springer.com/referenceworkentry/10.1007/978-3-642-20617-7. (accessed 20 June 2023).

De Schryver, G.-M. and D. Joffe. 2023. The end of lexicography, welcome to the machine: On how ChatGPT can already take over all of the dictionary maker's tasks: http://hdl.handle.net/1854/LU-01GWSGV0M8HYZVNXV8SSZ0VXBG.

du Sautoy, M. 2021. *Der Creativity Code. Wie Künstliche Intelligenz schreibt, malt und denkt*. München: C.H. Beck.

Embleton, S. and E. Wheeler. 1997. 'Multidimensional Scaling and the SED Data,' in *The Computer Developed Linguistic Atlas of England* 2, ed. W. Viereck & H. Ramisch. Tübingen: Max Niemeyer. 5-11.

Embleton, S. and E. Wheeler. 2000. 'Computerized Dialect Atlas of Finnish: Dealing with Ambiguity.' *Computer Science. Journal of Quantitative Linguistics* 7: 227-231.

Embleton, S., D. Uritescu and E. Wheeler. 2007. *Online Romanian Dialect Atlas*. http://vpacademic.yorku.ca/romanian (now at http://pi.library.yorku.ca/dspace/ under the 'dialectology' community, 'RODA' collection)

Eugster, J. 2017. *Übermorgen. Eine Zeitreise in unsere digitale Zukunft*. Zürich: Midas Verlag AG.

Frackiewicz, M. 2023. Reinforcement Learning: Teaching AI Through Trial and Error.' Artificial Intelligence, News on 27 April 2023. Cf. https://ts2.space/en/reinforcement-learning-teaching-ai-through-trial-and-error/.

Frege, G. 1879. *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. Halle: Verlag von Louis Nebert.

Gray, M. L. and S. Suri. 2019. *Ghost Work: How to Stop Silicon Valley from Building a New Global Underclass*. Boston, New York: Houghton Mifflin Harcourt.

Guanfu, S. 2020. 'What is AI GIS (Artificial Intelligence GIS)?' *Journal of Geo-Information Science*, 26 Feb 2020: https://www.supermap.com/en-us/news/?82_2701.html (accessed 7 Nov 2023)

Harari, Yuval Noah. 2019. *21 Lessons for the 21st Century*. London: Vintage.

Harold, E. R., and W. S. Means. 2004 (3rd ed.). *XML in a Nutshell*. Beyjing etc.: O'Reilly.

Herger, M. 2020. *Wenn Affen von Affen lernen: Wie künstliche Intelligenz uns erst richtig zum Menschen macht*. Kulmbach: Plassen-Verlag.

Hill-Yardin, E. L., M. R. Hutchinson, R. Laycock and S. J. Spencer. 2023. 'A Chat(GPT) about the future of scientific publishing.' *Brain, behavior, and immunity*, 110: 152-154.

Ing, L. Y. and G. M. Grossman (eds). 2023. *Robots and AI. A New Economic Era*. London, New York: Routledge.

Kirschfeld, K. 2020. 'Das schiefe Bild des Richard David Precht.' *Cicero: Magazin für Politische Kultur*, 6 Dec 2020. Cf. https://www.cicero.de/kultur/richard-david-precht-buch-kuenstliche-intelligenz.

Kortmann, B., K. Lunkenheimer, and K. Ehret (eds). 2020. *The Electronic World Atlas of Varieties of English*. Zenodo. Available online at http://ewave-atlas.org (last accessed 1 August 2023).

Markus, M. 2018. Review of Monika Wegmann. 2017. *Language in Space: The Cartographic Representation of Dialects*. Travaux de Linguistique et de Philologie. Strasbourg: Éditions de Linguistique et de Philologie. Rev. in *Anglia* 136 (3): 530-537.

Markus, M. 2021a. *English Dialect Dictionary Online: A New Departure in English Dialectology*. Cambridge: Cambridge University Press.

Markus, M. 2021b. 'OED and EDD: comparison of the printed and online versions.' *Lexicographica* 37 (1): 261-280.

Markus, M. 2022. 'A critical assessment of English dialect feature catalogues: Towards a dialectometrical evaluation of the English Dialect Dictionary Online.' *Lingua*, online publication first view 279:1-18.

Markus, Manfred. 2023. 'What is new in *EDD Online 4.0*?' *Dictionaries. Journal of the Dictionary Society of North America* 44 (1): 121-140.

Nerbonne, John. 2015. 'Various Variation Aggregates in the LAMSAS South.' In: Picone, Michael D. and Evans Davies, Catherine, eds. *Language Variety in the South: Historical and Contemporary Perspectives*. Tuscaloosa: University of Alabama Press. pp. 773-753.

Nerbonne, John, *et al.* (Wilbert Heeringa, Jelena Prokic, and Martijn Wieling). 2021. 'Dialectology for Computational Linguists.' In: Marcos Zampieri & Preslav Nakov (eds.) *Similar Languages, Varieties and Dialects: A Computational Perspective*. Cambridge: CUP. pp.96-117.

Onysko, Alexander. 2010. "Phrases, combinations and compounds in the English Dialect Dictionary as a source of conceptual metaphors and metonymies in Late Modern English Dialects." In *Joseph Wright's English Dialect Dictioary and Beyond. Studies in Late Modern English Dialectology*. Eds. Markus, M., C. Upton, and R. Heuberger. Berlin, etc.: Peter Lang. Pp. 1290-153.

Otte, R. 2021. *Allgemeinbildung Künstliche Intelligenz. Risiko und Chance für Dummies*. Weinheim: Wiley-Vch GmbH.

Precht, R. D. 2020 (3rd ed.). *Künstliche Intelligenz und der Sinn des Lebens*.München: Goldmann.

Rabanus, Stefan. 2018. 'Dialect Maps.' In Boberg, Charles, John Nerbonne, and Dominic Watt, eds. *The Handbook of Dialectology*. Hoboten, NJ.: John Weiley & Sons. Pp. 348-367.

Raggett, J., and W. Bains. 1992. *Artificial intelligence from A to Z*. London, etc.: Chapman & Hall.

Raphael, B. 1976. *The thinking computer – mind inside matter*. San Francisco, Ca.: W. H. Freeman.

Russell, S. J. 2019. *Human Compatible. Artificial Intelligence and the Problem of Control*. London: Allen Lane.

Simanowski, R. 2020. *Todesalgorithmus. Das Dilemma der künstlichen Intelligenz*. Wien: Passagen-Verlag.

Shane, J. 2021. *Künstliche Intelligenz. Wie sie funktioniert und wann sie scheitert*. Heidelberg: dpunkt. verlag.

Turing, A. M. 1950. 'Computing, Machinery and Intelligence.' *Mind* (59) 236: 433-460.

Viereck, W. and H. Ramisch, eds. 1997. *The Computer Developed Linguistic Atlas of England 2*. Tübingen: Max Niemeyer. Walmsley, P. see https://en.wikipedia.org/wiki/Artificial_intelligence (accessed 7 August 2023).

Walmsley, P. 2007. *XQuery*. Beijing etc.: O'Reilly.

Werner, P. 2022. 'Introduction to Map Layers for Backcountry Navigation'. https://sectionhiker.com/introduction-map-layers-backcountry-navigation/ (accessed 31 Oct 2023).

Yu, H. 2023. 'Reflection on whether Chat GPT should be banned by academia from the perspective of education and teaching.' *Frontiers in Psychology*, 14: 1181712.

Zell, A. 1994. *Simulation Neuronaler Netze* [Simulation of Neural Networks; in German]. Oldenbourg: De Gruyter.

Zeng, X. and D. S. Yeung. 2006. 'Hidden neuron pruning of multilayer perceptrons using a quantified sensitivity measure,' *Neurocomputing*, 69(7–9): 825–837.

Appendix: BaseX search for *maiden* as part of a combination

```
for $combTerm in (TEI//re[(@type="COMB")]/descendant::name/term) [matches(text(),"maiden","i")]
  let $comb:=$combTerm/ancestor::re[(@type="COMB" )]
  let $entry:=$comb/ancestor::entry
  let $compTermNum:=$combTerm/@n
  let $orth:=$entry/form/orth
  let $pos:=$entry/form/*/pos
  let $span:=$combTerm/preceding-sibling::span[1]
  let $spanValue:=replace($span,"[()|[]]","" )
  let $cit:=$comb/following-sibling::gb[1]/following-sibling::*[not(matches(name(),'lb'))][1][name()='dictScrap' and @change='CIT']
  let $spanCit:=$cit/descendant::span[@type="NUM"]
  let $spannext:=$spanCit[(replace(text(),"[()|[]]",""))= $spanValue][1]
  let $area0:=($cit/descendant::*[((preceding-sibling::span[@type="NUM"][1]/preceding-sibling::*[1][name()='span' and @type='NUM'])
[text() = $spannext/text()]  or ( preceding-sibling::span[@type="NUM"][1])[text() = $spannext/text()] )]  )
  let $area1:=$comb/descendant::*[preceding-sibling::name[1]/term[text() = $combTerm][1]]
  let $searchArea:=if(not($spanValue))  then ( $area1|$cit ) else ( $area1|$area0 )
  let $dialect:=$searchArea/(descendant-or-self::district | descendant-or-self::region | descendant-or-self::country)
  let $dialect3:=($comb/preceding-sibling::form[1]/(district | region | country)| $ comb/preceding-sibling::form[1]/* [not(matches(
name(),'usg'))] / descendant-or-self::title/(district | region | country))

 let $dialectExist:=( $comb|$cit)/( descendant::district | descendant::region | descendant::country )
  let $compPos:=$entry/descendant::name[(@type="COMB")][descendant::term[(@n=$compTermNum)]]/following-sibling::gramGrp[1]/pos
  let $etym:=$entry/dictScrap[(@change='COMMENT' )]/descendant::lang
  let $label:=$searchArea/(descendant-or-self::note[(@type, 'LABEL')]| descendant-or-self::note[( @type='GRAM')]| descendant-or-
self::gram  | descendant-or-self::m)
  let $years:=$searchArea/descendant-or-self::date
  let $prev:=string($entry/@prev)
  let $select:=string($entry/@select)
  let $title:=$searchArea/descendant-or-self::title
return

<result>
{
  $entry/(@n|prev|@select),
  $orth,$pos,
  if (not( $dialectExist))
  then
  (
      <compound>
      {
        <dialectFilter>{$dialect3}</dialectFilter>,
        $span,$combTerm,
        <labelFilter>{$label}</labelFilter>,
        <etymFilter>{$etym}</etymFilter>,
        <yearFilter>{$years}</yearFilter>,
        <timespanFilter>{<prev>{$prev}</prev>,<select>{$select}</select>}</timespanFilter>,
        <sourceFilter>{$title[term|country|district|region]}</sourceFilter>,
        $compPos
      }</compound>
    )
  else
   (
      <compound>
      {
      $entry/(@prev|@select),
      $span,$combTerm,$spannext,
      <dialectFilter>{$dialect}</dialectFilter>,
      <labelFilter>{$label}</labelFilter>,
      <etymFilter>{$etym}</etymFilter>,
      <yearFilter>{$years}</yearFilter>,
      <timespanFilter>{<prev>{$prev}</prev>,<select>{$select}</select>}</timespanFilter>,
      <sourceFilter>{$title[term|country|district|region]}</sourceFilter>,
      $compPos
      }</compound>)
     }

    </result>
```