Acta Linguistica Academica 69 (2022) 4, 599–614
DOI: 10.1556/2062.2022.00581

AKADÉMIAI KIADÓ

Principles of corpus querying: A discussion note

BÁLINT SASS[*]

Hungarian Research Centre for Linguistics, Institute for Lexicology, Hungary

Received: April 26, 2022 • Accepted: October 11, 2022
Published online: November 22, 2022

ABSTRACT

Nowadays, it is quite common in linguistics to base research on data instead of introspection. There are countless corpora – both raw and linguistically annotated – available to us which provide essential data needed. Corpora are large in most cases, ranging from several million words to some billion words in size, clearly not suitable to investigate word by word by close reading. Basically, there are two ways to retrieve data from them: (1) through a query interface or (2) directly by automatic text processing. Here we present principles on how to soundly and effectively collect linguistic data from corpora by querying i.e. without knowledge of programming to directly manipulate the data. What is worth thinking about, which tools to use, what to do by default and how to solve problematic cases. In sum, how to obtain correct and complete data from corpora to do linguistic research.

KEYWORDS

corpus, query, concordance, filter, frequency list, context

1. INTRODUCTION

Querying corpora seems to be a simple topic. Presumably, just write in some words in a textbox and receive the result. This does not work that way. The reason for this is that we do not look for specific information, but we want to get a relevant and complete linguistic dataset as a solid basis for our research. To be able to effectively query corpora you have to learn a suitable formal query language and other additional tools. This allows you to reveal all subtle details which are in the annotation of the corpus or in the text itself. This article is intended primarily for introductory or intermediate level readers to provide a solid foundation for the topic, but advanced readers will certainly find useful details in it as well.

[*] Corresponding author. E-mail: sass.balint@nytud.hu

There are quite a few excellent corpus lingustics handbooks (e.g. O'Keeffe & McCarthy 2010; McEnery & Hardie 2012; Weisser 2016) that provide a comprehensive description of the field. In this paper, we intend to complement these with a different perspective. Our objective is to look at the practice of corpus querying specifically and highlight general principles which are usually applied implicitly. The task is to state and formulate them in an explicit way so they can constitute a sound corpus querying mindset.

There will be eight principles in eight sections. Principles #1 and #2 elaborate on the two basic querying intention, namely searching for one perfect example on the one hand, and searching for all relevant data on the other. Principle #3 is likely to be surprising. It emphasizes the fact that not all annotation is perfect, and if we base our research on information annotated in a corpus (which is usually the case) we must be aware of the quality of the annotation. Principle #4 call your attention to the fact that different corpora used together can provide independent arguments to the investigation. Principle #5 presents the fundamental corpus-based research attitude (Tognini-Bonelli 2001). Principles #6 and #7 add technical details to principles #1 and #2. And finally, principle #8 brings querying to a new level taking the context into account.

Considerations about representativeness of corpora (Biber 1993) are outside the scope of the paper, we assume throughout that the corpora used are reprentative for our purposes. Similarly, we assume that the original raw text of the corpora are of good quality. The field of corpus cleaning is outside the scope as well. Hungarian examples will be provided. By glosses and explanations they will hopefully be understandable, and most importantly the essence and significance of the principles themselves will be clear.

The starting point for this article was (Sass 2017). The present study is a substantially extended and systematized English version of the principles published there in a rudimentary form or just as ideas.

1.1. Concepts and terminology

A "raw" corpus is a series of tokens which are either word forms or punctuation marks. In contrast, an "annotated" corpus contains some annotations assigned to individual tokens or to spans of tokens. One of the most common annotations is the lemma which assigns its stem to every token.

A central concept of corpus querying is "concordance" (Sinclair 1991). This is the result of a query. This is a list of "hits" (incidences). A hit consists of the queried word or expression called "kwic" (keyword in context) and a variable size context of it.

Token positions are counted from the kwic: one token to the left, two tokens to the left and one token to the right are `−1`, `−2` and `1` respectively. A "window" is a span of tokens defined by a starting and an ending position, e.g. a window denoted by `−1…2` consists of four words: the kwic, one to the left and two to the right. Frequency lists are created not only from kwics (i.e. position `0`) but from tokens appearing at other positions.

A corpus query system (CQS) is a system which allows the user to create concordances from corpora by running queries.

1.2. Tools

We will assume that our CQS is the NoSketchEngine (NoSkE) (Rychlý 2007). Our main tool will be its formal query language called CQL (Corpus Query Language) (Sketch Engine Team 2015). The reader should be familiar with CQL and regular expressions as well to understand the examples in this paper.

A formal query language has to be capable of querying at character level and also at token level. The essence of CQL is that it meets this requirement by using the concept of regular expressions at both levels. Example (1) illustrates this.

(1) a. `[word = “m.+a”]`

b. `[word = “egy”] []+ [word = “munka”]`

Query (1a) searches for words beginning with *m* and ending with *a*, while (1b) searches for expressions in which the first word is *egy* and the last is *munka*. The pattern expressed by a regular expression is the same at both levels: a certain sign, then anything (`.` and `[]` respectively) one or more times (+), then another certain sign. This nicely reflects the double articulation of language.

The universal codes used in this paper are the following: attribute `word` is for word form, `lemma` is for lemma and `pos` is for part of speech and morphological features. Values for the latter are in Table 1. To try the queries presented in this paper for a certain corpus you should adjust the codes to the actual annotation of the corpus.

Beyond the formal query language it is worth to be familiar with the two basic operations of corpus querying: filtering and frequency list creation. When querying corpora you should always keep in mind that (1) you have the possibility to break down a complicated problem into steps and use several queries, one after another applied to the result of the former; and (2) it is useful to create a frequency list from almost every single query as this shows what is typical and what is not. Certainly, all filtering series can be formulated as a single complicated query, but it can be useful to divide a complex task into parts instead. Of course a filtering step can be done using a CQL query. More details about this will be covered in principles #6 and #7.

2. THE PRINCIPLES

#1 Principle of examples

One of the excellent features of corpora is that they are the authentic source of real, living linguistic examples. Instead of using hand-crafted examples for linguistic phenomena we

Table 1. Universal part of speech codes used are in first column. Specific codes for running queries directly in HGC v2.0.5 are in third column. In HGC, the name of the attribute is `msd` instead of `pos`.

universal code	meaning	HGC code
`DET`	determiner	`DET`
`NOUN`	noun	`FN.*`
`NOUN.ACC`	noun in accusative case	`FN.*ACC`
`ADJ`	adjective	`MN.*`
`VERB.DEF`	verb with definite conjugation	`.*IGE.*T.*`
`INF`	infinitive	`IN[FR].*`

encourage researchers to build linguistic research at least in part on data from corpora. In contrast with sterile, hand-crafted examples which can be more on the point, corpus examples may be more fuzzy, but they are more typical and show better how language works in practice.

Thinking of a good example sentence for *megkerül* ‘get.around’ what first comes to mind is something like *Jóska megkerülte a házat.* ‘Joe went.around the house.’ However, taking a look into the corpus sentence (2) emerges.

(2) *Nem akarom meg-kerülni a kérdést.*
not want.1SG around-get.INF the issue.ACC
‘I don’t want to ignore the issue.’

This shows that the figurative meaning is more typical together with auxiliary *akar* ‘want’, first person singular and negation.

For expression (3) sentence (4) could be a fine example at first sight.

(3) *helyzetbe hoz*
position.INE put
‘put in position’

(4) *Helyzetbe hozta a csatárt.*
position.INE put.3SG the striker.ACC
‘He put the striker in position.’

The Hungarian Gigaword Corpus (HGC) (Oravecz, Váradi & Sass 2014) reveals that sentence (5) is far more suitable.

(5) *Az infláció rendkívül nehéz helyzetbe hozza az önkormányzatokat.*
the inflation extremely difficult position.INE put.3SG the local.governments.ACC
‘Inflation puts local governments in an extremely difficult position.’

This does not just show that the figurative meaning is typical again, but also that there is an adjective and an adverb as well in this expression quite often. Moreover, the adjective typically has a negative meaning (e.g. *nehéz* ‘difficult’, *lehetetlen* ‘impossible’, *kellemetlen* ‘unpleasant’), and the adverb an enhancer meaning (e.g. *rendkívül* ‘extremely’, *nagyon* ‘very’, *eléggé* ‘quite’). Note that the second example demonstrates all these typical elements while the first one does not.

It is not necessary to take over examples from corpus fully accurately, i.e. we can omit parts which are not relevant for some reason. In (5) we omitted an adjective phrase, instead of (6) we used *infláció*inflation simply keeping the necessary components of the expression intact.

(6) *az Energia és más területeken meglévő infláció*
the Energy and other area.PL.SUP existing inflation
‘inflation in Energy and other areas’

Such examples are called “corpus-based examples”. They are used in dictionary definitions often.

The above thought process shows as well, how the so called serendipity property of corpus research works in practice (Wilkinson 2007). That means taking a look into the corpus often sheds new light on the original issue (often formulated at a theoretical level), adds new aspects, shapes or modifies it, and raises new questions and even new research directions by bringing in a practical language usage perspective. This is an important reason why we encourage using corpora in linguistic research.

To obtain a good example, the first step is to define the most specific surface element of the phenomenon we examine and query for that element. Then apply the principles presented in the following sections, most of all, filtering (see principle #6). The corpus can always provide a typical context of the phenomenon.

#2 Principle of all hits

When we search in a corpus for a certain linguistic phenomenon, we can basically have two kind of goals. Either we would like to have some examples (see principle #1) or we would like to have all incidences. In other words, the goal is either only to demonstrate the phenomenon or perform a quantitave analysis on it. In the first case, the precision of the query is important, this allows us to obtain appropriate examples quickly, without spending much time sorting the hits. In the second case, high recall is required, because a quantitative investigation should take all relevant data into account, even at the price of having to (manually) discard irrelevant hits. In this section we cover the second case.

This principle states: all hits of a phenomenon are necessary to be able to perform an reliable quantitative analysis.

A simple consideration can be whether to search for a word form or for a lemma. For example, investigating expressions like (7) the verb should be queried as lemma, since the verb can appear in any conjugated form, and the other two words as word form, since they are fixed in this expression.

(7) *fel-kap-ja* *a* *vizet*
up-pick-3SG the water.ACC
‘become angry’

The appropriate query can be seen in (8a). Using query (8b), with the third person singular form of the verb in it, we would lose even 60% of the hits.

(8) a. `[lemma=“felkap”] [word=“a”] [word=“vizet”]`

b. `[word=“felkapja”] [word=“a”] [word=“vizet”]`

To put it somewhat differently, the task of corpus query systems is to provide all hits the user “thinks of” while using the system. But as long as we do not have such smart linguistic search engines, the user must think about how to obtain really all hits. We do research of specific linguistic phenomena, so simply typing a word, pressing enter and waiting for the perfect result (aka “googling”) does not work (Kilgarriff 2007). The corpus cannot and will not automatically answer our research question in the vast majority of cases (see principle #3).

Including the above consideration, the following are worth thinking about:

1. Lemma versus word form as detailed above.
2. Misspelings. If there is a significant amount of for example *hoyg* instead of *hogy* 'that' then our numbers can become inappropriate.
3. Spelling variants. Examining a raw Old Hungarian Corpus (Simon 2014), querying *majd* 'then' is not enough, you should take care of all spelling variants, e.g. *maÿd.*
4. Dialectal variants. There is a phenomenon in Hungarian which had become common in standard language: the alternation between *e* and *ö*. If we search for *tejföl* 'sour cream' and forget about *tejfel* we will lose more than 10 percent of the data.
5. Word form variants. The researcher may consider including not only *maÿd* but also *majdan* (same meaning) in his data to be investigated.
6. Synonyms. There can be scenarios when even synonyms are to be included, i.e. if they are to be treated the same for some reason.

These kind of features can be annotated in a corpus. If so, we have an easy task. Lemma is the most commonly annotated feature in an annotated corpus, misspellings can be corrected using spell check tools, and spelling variants in old texts can be turned into the corresponding synchronic form by the process of normalization. The other three features are very rarely annotated.

Generalizing the concept, we can say that the purpose of normalization is to project all corpus tokens to the query which can be projected. In other words, normalization is the process of bringing two different tokens to the same representation, if the difference between them is "linguistically irrelevant".

In this sense, processes which solve the above six problems are normalization processes. For example, lemmatization is normalization as it projects all inflected forms to the lemma (e.g. *felkapja, felkapná… → felkap*), and annotating synonyms is also a kind of normalization as it brings all synonymous tokens to a common representation, e.g. a WordNet (Miháltz et al. 2008) synset.

But how can we decide what is "linguistically relevant"? How can we figure out the intentions of the user? There is no general solution, since what difference is relevant is completely research-dependent. Differences which are crucial for the research of historical change in spelling are irrelevant for research of syntax.

Knowing that any kind of difference can be relevant, it is important to always store pre-normalization and post-normalization representations both. Thus different "levels" of the text come into being, starting from the original word forms. The original raw text should be always present, because this is the representation which contains all differences which are present in the corpus.

It is an open question whether there can exist a general solution to the problem of normalization which allows us to be able to decide what is relevant for our actual research not during building the corpus but during querying the corpus.

#3 Principle of imperfection

More data makes results more reliable and allow rarer phenomena to be investigated properly. Accordingly, corpus linguistic research is often done on really large corpora nowadays whose size can be even several billion (10^9) words. One billion word text can be imagined as a 500 m long bookshelf full with books.

Of course, it would be best to work with flawless data. The situation is the following: such huge amount of data cannot be annotated manually, only automatically. Automatic natural language processing methods cannot be perfect (100 percent accurate) due to the “fuzzy” nature of language. Accordingly, while small, manually created corpora can be flawless aka gold standard, automatic corpus annotations in large corpora are necessarily somewhat defective. We need to be aware of this fact and we have to do something with it. The most important thing is not to believe that corpora are flawless. Do not blindly trust the annotation of the corpus.

The biggest problem with the annotation being wrong is that recall decreases, therefore some relevant data cannot be obtained making it impossible to fulfill principle #2. For example in Hungarian *terem* can mean ‘hall’, ‘produce/grow’ or ‘space.my’. If POS tagging and/or lemmatization is flawed the above problem emerges.

Let us be aware of how much confidence we can have in the annotation. Get a random sample from the corpus which exemplifies the assumed defectivity. Then check it manually how many of the hits are wrong. Include this ratio in the publication and either draw attention to the fact that the results are to be interpreted knowing this, or state that, because of that, we have discarded the original approach and applied something else, e.g. querying the raw corpus in a tricky way.

If an annotation task is hard for a human, suspect that it may be hard for a machine as well. Do not expect perfect annotation in a difficult task. For example:

1. Word sense disambiguation: Hungarian *barát* can mean ‘friend’ or ‘monk’. As they are both nouns, the task is much harder than the above concerning *terem*.
2. In Hungarian, as in English, past participle and past tense third person singular form (*elkészített* ‘prepared’) of a verb (*elkészít* ‘prepare’) coincide. This is a classic example which is often erroneous in the HGC.
3. The definite counterpart of the above indefinite example is even more problematic as the past participle version is much rarer than the verb form version. E.g. *ember alkotta csodák* ‘man-made miracles’ versus *ember alkotta a csodákat* ‘man made the miracles’ (Kenesei 1986). Note that as this phenomenon is quite rare and we wanted to present a typical example, the above example was collected applying principle #1 using CQL query (9).

(9)
```
[pos=“NOM”]  [word=“.+tt[ae]”]  [POS=“NOUN”]
```

The above examples are from the field of word sense disambiguation and POS tagging, but clearly, other types of annotation can be even more challenging: e.g. annotation of sentiment, metaphor, irony or euphemism. High level tasks of this kind are often impossible even manually to be annotated without any error.

Not only do not expect perfect annotation, but even think about whether the available annotation can anwser our research question at all. Never expect that the complete answer to our very research question is readily and perfectly annotated in the corpus. In other words, do not try to burden the corpus with the responsibility for solving the research task itself. The solution is usually not automatically there, but the data which helps often is, we just have to pull it out from the corpus somehow, e.g. using the principles described in this paper. On the other hand, it is good practice to annotate – automatically or manually – the researched phenomenon in a corpus (see Vadász 2020) and make it available to others to help further research.

Try to use the available annotation creatively. You can even completely ignore the annotation, and query the corpus as if it was a raw corpus. In other words, if there is no other idea we should go back to the original text, to the raw corpus. For example, if you want to find basic determiners in an English corpus you can use query (10a) instead of query (10b) if you notice that the set of words annotated as `DET` is much larger than what you want to get.

(10) a. `[word="a|an|the"]`

b. `[pos="DET"]`

As the consequence of imperfection of annotation is that it is worth developing skills to query raw text, this principle can also be called “principle of raw text”.

To sum up this section, it is a good idea to follow the cautious attitude of not expecting the annotation to be perfect, not expecting that the corpus will automatically provide solution to our research. Try to query the available annotation creatively to obtain data which is needed instead.

#4 Principle of cooperation

If a very good corpus is available, it can happen that one gets used to always using it. This can be a bad practice. Consider using different corpora for different research, or even better, using multiple corpora for the same research. Each corpus adds its own perspective, complementing each other.

It makes the work more efficient and more convenient if all corpora are available in the same corpus query system. We argue that NoSkE can be a good unified platform for querying corpora. NoSkE is free of charge, well known, fast even at billion-word-sized corpora, has full CQL support, has filtering and sophisticated frequency list generation functionality, and accompanied with a helpful email list for users.

To do sound corpus linguistic research one general purpose corpus is not enough, several special corpora are also needed beyond that. A so called “basic corpus set”.

In Figure 1 a three dimensional basic corpus set can be seen. This set consists of four corpora: a general purpose corpus (in the middle), others covering child, spoken and historical language. Three dimensions or aspects are usually sufficient, but other special corpora can also

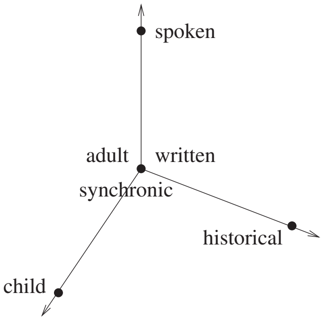

Figure 1. A basic corpus set with three dimensions or aspects

be inserted into the system. It is enough if only the general purpose corpus is large, a smaller corpus of a given aspect is sufficient for examinations concerning this aspect.

For Hungarian, the following corpora constitutes such a basic corpus set. Hungarian Gigaword Corpus (HGC) (Oravecz, Váradi & Sass 2014) is a general corpus, Hungarian Kindergarten Language Corpus (HUKILC) (Orosz & Mátyus 2014) contains child language, Budapest Sociolinguistic Interview (BSI) (Kontra & Váradi 1997) is a spoken corpus available also in transcribed form and Hungarian Historical Corpus (HHC) (Pajzs et al. 1998) is a historical corpus.

We have said at principle #3: do not expect that the answer to our research question is annotated in the corpus. Here we add that another corpus may very well contain the answer you need.

Now, we present a longer case study, a simple investigation which shows the power of cooperation of corpora along with applying other principles as well. The word list (11) were presented on a slide at a conference as Hungarian words borrowed from Romani (Kresztyankó 2016).

(11) csaj, csávó, csór, gádzsó, gizda,
góré, kaja, kéró, lóvé, nyikhaj,
pia, pimasz, séró, verda

At first glance, one was completely out of line for me: *pimasz* ‘cheeky’ (as in cheeky *kid*) seemed to be a very old Hungarian word. To find out, I checked first occurrences ot these words in the HHC.

The result in Figure 2 clearly shows that *pimasz* is out of line: it can be a Romani loanword, but the corpus clearly shows that it has been present in Hungarian for a much longer time compared to the others. This fact can be an expanation for why a Hungarian native speaker feels that this is a completely colloquial/neutral word. It can be noted that *csór* ‘steal’ also feels to be a bit out of line and the figure supports this feeling as well. It turns out that this investigation did indeed give a correct result as it is verfiable that *pimasz* is in fact not a Romani loanword (Turner 1962–1966).

To establish how much *pimasz* is colloquial/neutral, we can turn to a large general purpose corpus which provides more accurate frequency data due to its size. The idea is the following: frequency of word compared to the closest standard synonym.

Table 2 shows that frequency of *pimasz* is the same as its standard closest synonym, while that of *csaj* ‘girl’ is much more lower. That supports the claim that *pimasz* and *szemtelen* can be

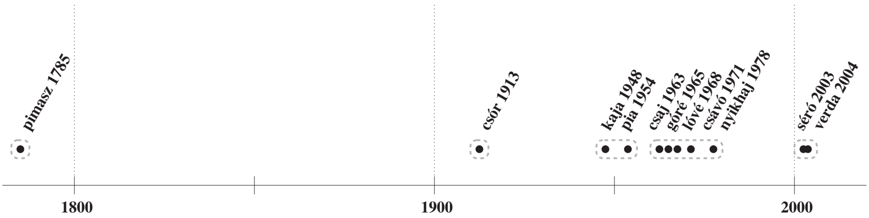

Figure 2. First occurrences of words assumed to be of Romani origin in the Hungarian Historical Corpus

Table 2. Frequency ratios of two loanwords and their standard closest synonym counterparts

meaning	loanword	#	standard	#	frequency ratio
‘cheeky’	*pimasz*	1825	*szemtelen*	1976	=
‘girl’	*csaj*	10000	*lány*	198000	×20

freely interchanged, but *csaj* and *lány* cannot, the latter word pair has a significant difference in stylistic value and consequently in usage. So this small investigation supports our hypothesis that *pimasz* is a colloquial/neutral word.

Lány means ‘girl’ while *csaj* is something like ‘chick’. The fact that *csaj* also has frequently used derivative forms supports that this word is smoothly integrated into the Hungarian language. Beyond the adjective *lányos* ‘girly’ we have *csajos* ‘be like a chick’ as well. Their different collocation patterns also show the meaning and usage differences of these two words. While (see principle #1) *lányos* have long-standing more idiomatic collocations like (12), *csajos* have literal ones like *csajos film* ‘chick flick’ or *csajos buli* ‘party for chicks’.

(12)	*lány-os*	*apa*
	girl-y	father
	‘father having daughters’	

The above investigations and the data in Table 2 are in line with the sense of language of the native speaker concerning “how foreign does a word seem”. *Pimasz*, *csaj* and e.g. *gádzsó* ‘man’ are at the beginning/middle/end of the foreignness scale from all three perspectives. Using two corpora in cooperation allowed us to confirm our original hypothesis. Always keep in mind that you can use more than one corpus.

#5 Principle of first thought

Corpora, as a source of linguistic data, serve to support or refute linguistic conjectures and hypotheses. If we come across a linguistic statement, then if the appropriate corpus is available, we can immediately check the truthfulness and correctness of the statement. An attitude, a way of thinking, can be developed so that when such a claim or question arises, we reach out to the corpus right away, as an evident skill, in a natural way, and look for an answer there. Our first thought should or at least can be: let us check this in a corpus.

For example, if the question is “how long has a specific word been in Hungarian?” checking a historical corpus you get the answer at least in the sense that you get a date from which this word is surely present in Hungarian. As another example, this claim can also make you think: “in Hungarian, comparative (i.e. *-bb*-suffixed) forms of adjectives always comes with *-a-/-e-* linking element except only a single adjective: *nagy* ‘big’ where the linking element is *-o-*” (Kálmán 2011). Is this true? The first thought is to check this in a large general purpose corpus by an appropriately crafted query. Investigating the frequency list created from query (13) will show us that the single comparative form in the is *nagyobb* ‘bigger’ so the statement is true indeed.

(13) `[word=‟.+obb″]`

The significance of this principle goes far beyond querying corpora, it is crucial in solving numerous other kinds of problems. The information about whether a phenomenon exists (see principle #1) or how frequent it is (see principle #2) is often the basis of certain decisions.

For example, a spell checker can be improved using a corpus: we can accept a word as correct not only when the spell checker accepts it but also when the word is present (in an appropriate amount) in a clean corpus. Or for the task of correcting OCR errors a useful approach can be to check the frequency of a given word in the OCR-ed text and also in a clean corpus, and if the frequencies are roughly equal, then the word probably does not contain any OCR-errors even when other methods find it suspicious.

Obviously, principle of first thought is the most fundamental principle of corpus querying. All other principles are based on this. Principle #1 and principle #2 show the two essential approaches of querying, principle #3 warns us to be careful, principle #4 takes us further to using multiple corpora at the same time. After two more technical principles about basic operations of querying principle #8 will show the real power of querying.

#6 Principle of filtering

This principle is about the approach of narrowing down the set of hits step by step, using several queries one after another to get the desired dataset. Or put it another way, your query does not have to be perfect at first. If the results are not specific enough, you can use further queries on the result of the former to have a more appropriate dataset. This is called “filtering”.

Take a look at the problem of obtaining Hungarian past tense verbs ending with *-tt* in raw text. Query (14a) will not be sufficient as there are several other words with this ending. The solution is filtering by a query something like (14b) to exclude non-verbs.

(14) a. `[word=‟.+tt″]`

b. `[word!=‟itt|ott|alatt|miatt|között|mellett″]`

In NoSkE, filtering can be applied for a window. We used a `0..0` window in the above filtering step, as we wanted to apply it to the kwic itself.

Another example is searching for Hungarian verb–preverbs constructions. Here we formulate our query to use filtering from the beginning. In Hungarian, the preverb can be written together (15) or separately (16) from the verb to which it belongs.

(15) *át-megy*
through-go.3SG
‘goes through’

(16) *nem* *megy* *át*
not go.3SG through
‘does not go through’

When written separately, the preverb can go far from the verb, but the overwhelming majority of preverbs are at most 3 words to the left and at most 2 words to the right. To obtain all these hits (see principle #2) of a preverb-verb combination we can apply filtering as a good solution.

After querying the verb by (17a) we apply filtering in a −3…2 window by querying the preverb using (17b).

(17) a. `[lemma="megy"]`

b. `[word="át"]`

To obtain forms written as one word as well, we can query `[lemma="átmegy"]` and take the union of this query and the former one. Note that union is the opposite process compared to filtering.

In order to obtain good examples (see principle #1), we basicly run a query and manually choose a suitable example manually. This simple method is satisfactory, but if we get a lot of hits, we have to decrease their number to a manageable amount to choose from. The most important method to decrease the number of hits – and obtain better examples at the same time – is filtering. If we have already applied all filtering ideas, but there are still to many hits, we can use random sampling to have a manageable amount of data. This functionality is provided by NoSkE out of the box.

A practical technique concerning filtering is "hit minimization" which is intended to speed up queries. If you have an option, choose a query which provides fewer hits. For example, we want to search for DET + ADJ patterns. If there are more determiners in the corpus than adjectives – which will be the case indeed –, query for adjectives first `[pos="ADJ"]` and then filter by determiners `[pos="DET"]` in a −1…−1 window which specifies the immediately preceding position, not the other way around. The technique of querying the most specific surface element mentioned under principle #1 is an application of hit minimization.

#7 Principle of focus

When we want to perform a quantitative analysis (see principle #2) on a certain linguistic phenomenon, then this phenomenon is our focus. Our query has to be formulated so that the focus will be in a position which is suitable for creating a frequency list from it, as this is usually the first step in a quantitative analysis.

A trivial example would be the following. To investigate the distribution of different inflected forms of e.g. *munka* 'work' a simple `[lemma="munka"]` query is adequate. Then, creating a frequency list of the forms gives the answer.

In Hungarian the future tense is formed by the auxiliary *fog* and the infinitive of the verb. These two elements can be placed quite far from each other. A task can be: create a frequency list of infinitives 1, 2 or 3 words apart from *fog* in any direction. The point is that the focus is the infinitives in this case as they should be the subject of frequency list creation. Accordingly, the first query has to be (18a) and then can come a filtering step by applying (18b) in a −3…3 window.

(18) a. `[pos="INF"]`

b. `[lemma="fog"]`

This is the order of queries which allows you to solve the task, the reverse order is not.

Note that this principle is intertwined with principle #6, as usually filtering is what allows us to grab the focus. In the above example querying the whole expression in one using (19) loses focus, because thus only the whole expression can be the subject of a frequency list.

(19) `[pos="INF"] []{0,2} [lemma="fog"]`

However, if the focus of the investigation is the whole expression, this is the way to go. But using this approach, frequency lists from certain elements of the expression cannot be created. In other words, the expression can only be investigated as a whole, not word by word. The lesson is to always think about which approach is suitable in a particular quantitative analysis: querying one word and filtering with others or querying the whole expression in one.

Note also that this principle may also conflict with principle #6, namely with hit minimization, as the requirement for focus can clearly override the other principle.

So far, we always created frequency lists from the kwic. An especially useful feature of NoSkE is the possibility of creating a frequency list from a token at a certain position relative to the kwic. This feature will be explained in more detail in the next section.

#8 Principle of context

This principle is often applied incidentally, but it is important to emphasize it on its own. The idea can be formulated very simply. Basically, we query for a word by its form or the information in its annotation. If this approach does not work, use form or annotation information of context words. This is a general idea that comes in many forms, as the following case studies show.

1. Do you want to know which adjectives collocate with *munka* 'work' or whether a particular adjective is typical? You can (1) search for this word as lemma `[lemma="munka"]`, (2) filter (see principle #6) the concordance for adjectives `[pos="ADJ"]` in the immediately preceding position by window `−1…−1`, and (3) create a frequency list (see principle #7) from these adjectives, i.e. from the tokens at the `−1` (1 to the left) position. From HGC, you get *közös* 'joint', *szakmai* 'professional', *jó* 'good', *kemény* 'hard', etc. We can say, we used *munka* here as a context to find the adjectives in question.
2. Let us look at the following more general and more difficult problem. How to query Hungarian adjectives (in nominative case) in a raw corpus, e.g. in HHC? We have no annotation on which we could rely, and also, as Hungarian adjectives have no special ending, we have no clue in the form of the token either. The idea is this: many such adjectives would be found immediately before nouns, so use the context. Relying on specific case markers (e.g. by *-ban/-ben* 'in', see (20)) we can found nouns with high precision and then create a frequency list of tokens at `−1` position.

(20) `[word=".+b[ae]n"]`

The result shows that our approach is unsatisfactory: we obtain a lot of determiners, conjunctions, adverbs, etc. beyond adjectives this way. But we can go further and use another piece of context. As `DET` + `ADJ` + `NOUN` is a common noun phrase structure, we can filter the result requiring a Hungarian article (*a/az* 'the' and *egy* 'a/an') at −`2` position (i.e. two words to the left counted from the noun), by query (21).

(21) `[word="a|az|egy"]`

Creating a frequency list of tokens at −`1` position (still counted from the noun) we obtain adjectives with high precision, for example: *első* 'first', *nagy* 'big', *magyar* 'Hungarian', etc. Interestingly, using this approach we disambiguate words which can be both noun and adjective (e.g. *szomszéd* 'neighbour'), and accept adjectives with ortographic errors as well, as the context ensures that anything is this context must be an adjective in nominative case. What we get is a list of typical examples (see principle #1) but clearly not all adjectives (see principle #2). If we have no more appropriate approach to collect all adjectives, we can use this result as "all hits", but this fact should be noted in the investigation.

3. An even more complex task would be to search for positive adjectives. The solution is also complex, beyond the CQS it uses external tools as well, but the idea behind it is the same: use the context. The method could be the following. (1) Query for *jó* 'good' and (2) get a frequency list of the immediately following token (+`1` position) and save it. (3) Query for *rossz* 'bad' and do the same. (4) Subtract the second list from the first by some external tool (e.g. Excel), in other words, produce a list which contains words that can appear after *jó* 'good' and cannot appear after *rossz* 'bad'. If this list is ready, (5) query the words in it one by one and (6) get a frequency list of their immediately preceding (−`1` position) tokens. These words will be positive adjectives with a high precision. For example, from *akarat* 'will' we get *szabad* 'free', *szent* 'sacred' and *szilárd* 'solid'; or from *hazafi* 'patriot' we get *igaz* 'true', *lelkes* 'enthusiastic' and *hű* 'faithful' querying the HHC. The method outlined here can be called "context of context" querying.
4. Some phenomena have a very specific property that they often accumulate in a listing. In this case, we can query a known example, look at the the surronding listing, collect new examples from the context, and then iterate this method until we do not find new examples. An example of such a phenomenon is *villany lekapcsol* 'light switches.off', where the interesting feature is that the object is used in subject form (Halm 2021). This method can be called "listing as context" querying.
5. Finally, we may also be searching for something that is not there. The definite article is often omitted in Old Hungarian, where it is present in modern Hungarian. How to search for these articles which are not there? Our only possibility to solve this task is to rely on the context. Namely, specify contexts where the definite article should be there. Such a context is: verb with definite conjugation + definite article + noun in accusative case. Thus, running (a suitably adapted form of) query (22) on the Old Hungarian Corpus (Simon 2014) we obtain articles that are not there between the two words.

(22) `[pos="VERB.DEF"] [pos="NOUN.ACC"]`

This last mentioned method is "context of absence" querying.

3. CONCLUSION

My experience shows that beginners tend to use corpus query systems the way they use internet search engines. They tend to expect automatic answers. This does not work that way. Firstly, our task is not searching for a specific piece of information, i.e. information retrieval, but collecting relevant data – often all possible relevant data – about a specific linguistic research question, secondly, we research original problems which are obviously essentially never directly annotated in the corpus.

There is no free lunch, we should think about how to obtain the data we need. Principles presented in this paper are to support this thinking. The scheme how these principles work together is the following. It is worth getting used to it.

Imagine that we have a research question about a linguistic phenomenon. The first thought is: let us see what corpus data tells us about it (principle #5). Do we need some examples only (principle #1) or all relevant data for a quantitative analysis (principle #2)? After we have checked that the annotation we build on is good enough (principle #3), formulate the query, or more often the series of queries using filtering (principle #6), locate the focus and start the analysis with creating some frequency lists (principle #7). If we encounter difficulties, think about whether using more corpora brings us closer to the answer (principle #4), and whether using information from the context of the phenomenon in question helps (principle #8).

Having the eight principles of corpus querying in mind will hopefully help you to find the data which is needed for your research, and which is usually "right in front of you", you just have to have the methods and tools to unfold it from the corpus.

ACKNOWLEDGEMENT

This paper is the essence of my lecture on corpus querying which I held at several universities for years. I am grateful to the audience for their comments.

REFERENCES

Biber, Douglas. 1993. Representativeness in corpus design. Literary and Linguistic Computing 8(4). 243–257.

Halm, Tamás. 2021. Radically truncated clauses in Hungarian and beyond: Evidence for the fine structure of the minimal VP. Syntax 24(3). 376–416.

Kálmán, László. 2011. A nyitótövekről [On opening stems]. Nyelv és Tudomány. https://www.nyest.hu/hirek/a-nyitotovekrol.

Kenesei, István. 1986. On the role of the agreement morpheme in Hungarian. Acta Linguistica Hungarica 36(1–4). 109–120.

Kilgarriff, Adam. 2007. Googleology is bad science. Computational Linguistics 33(1). 147–151.

Kontra, Miklós and Tamás Váradi. 1997. The Budapest sociolinguistic interview: Version 3: Working Papers in Hungarian Sociolinguistics 2. Budapest: Linguistics Institute, Hungarian Academy of Sciences.

Kresztyankó, Annamária. 2016. Cigány jövevényszavak nyelvtanórán [Romani loanwords in grammar lessons]. In Sz. Bagyinszki and R.P. Kocsis (eds.) Anyanyelvünk évszázadai, Vol. 2. Budapest: ELTE BTK. 215–223.

McEnery, Tony and Andrew, Hardie. 2012. Corpus linguistics: Method, theory and practice. Cambridge: Cambridge University Press.

Miháltz, Márton, Csaba Hatvani, Judit Kuti, György Szarvas, János Csirik, Gábor Prószéky and Tamás Váradi. 2008. Methods and results of the Hungarian WordNet project. Proceedings of the Fourth Global Wordnet Conference. 311–321.

O'Keeffe, Anne and Michael McCarthy (eds.). 2010. The Routledge handbook of corpus linguistics. London: Routledge.

Oravecz, Csaba, Tamás Váradi and Bálint Sass. 2014. The Hungarian Gigaword Corpus. Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC2014). 1719–1732.

Orosz, György and Kinga Mátyus. 2014. An MLU estimation method for Hungarian transcripts. In P. Sojka, A. Horák, I. Kopeček and K. Pala (eds.) Proceedings of TSD 2014. Cham: Springer. 173–180.

Pajzs, J. et al. 1998. Magyar történeti szövegtár [Hungarian historical corpus]. http://www.nytud.hu/hhc/.

Rychlý, Pavel. 2007. Manatee/Bonito – a modular corpus manager. Proceedings of the 1st Workshop on Recent Advances in Slavonic Natural Language Processing. 65–70. https://nlp.fi.muni.cz/trac/noske.

Sass, Bálint. 2017. Keresés korpuszban: A kibővített Magyar történeti szövegtár új keresőfelülete [Querying corpora: The new search interface of the expanded Hungarian historical corpus]. In T. Forgács, M. Németh and B. Sinkovics (eds.) A Nyelvtörténeti kutatások újabb eredményei IX. Szeged: Department of Hungarian Linguistics, University of Szeged. 267–277.

Simon, Eszter. 2014. Corpus building from Old Hungarian codices. In K.É. Kiss (ed.) The evolution of functional left peripheries in Hungarian syntax. Oxford: Oxford University Press. 224–236.

Sinclair, John McHardy. 1991. Corpus, concordance, collocation. Oxford: Oxford University Press.

Sketch Engine Team. 2015. CQL – Corpus Query Language. https://www.sketchengine.eu/documentation/corpus-querying.

Tognini-Bonelli, Elena. 2001. Corpus linguistics at work. Amsterdam & Philadelphia, PA: John Benjamins.

Turner, Ralph Lilley. 1962–1966. A comparative dictionary of Indo-Aryan languages. Oxford: Oxford University Press. https://dsal.uchicago.edu/dictionaries/soas.

Vadász, Noémi. 2020. KorKorpusz: kézzel annotált, többrétegű pilotkorpusz építése [KorKorpusz: Construction of a hand-annotated, multi-layer pilot corpus]. In G. Berend, G. Gosztolya and V. Vincze (eds.) Proceedings of the 16th Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2020). Szeged: Institute of Informatics, University of Szeged. 141–154.

Weisser, Martin. 2016. Practical corpus linguistics: An introduction to corpus-based language analysis. Malden, MA, Oxford & Chichester: Wiley Blackwell.

Wilkinson, Michael. 2007. Corpora, serendipity & advanced search techniques. The Journal of Specialised Translation. https://jostrans.org/issue07/art_wilkinson.php.