From built examples to attested examples: a syntax-based query system for non-specialists

Ilaine Wang, Sylvain Kahane, Isabelle Tellier

From built examples to attested examples: a syntax-based query system for non-specialists

Ilaine Wang[1,2] **Sylvain Kahane**[1]
[1]MoDyCo (UMR 7114), CNRS
Université Paris Ouest Nanterre La Défense
i.wang@u-paris10.fr
sylvain@kahane.fr

Isabelle Tellier[2]
[2]LaTTiCe (UMR 8094), CNRS, ENS Paris
Université Sorbonne Nouvelle – Paris 3
PSL Research University, USPC
isabelle.tellier@univ-paris3.fr

Abstract

Using queries to explore corpora is today routine practice not only among researchers in various fields with an empirical approach to discourse, but also among non-specialists who use search engines or concordancers for language learning purposes. While keyword-based queries are quite common, non-specialists are less likely to explore syntactic constructions. Syntax-based queries usually require the use of regular expressions with grammatical words combined with morphosyntactic tags, meaning that users need to master both the query language of the tool and the tagset of the annotated corpus. However, non-specialists such as language learners may prefer to focus on the output rather than spend time and efforts mastering a query language. To address this shortcoming, we propose a methodology including a syntactic parser and using common similarity measures to compare sequences of automatically produced morphosyntactic tags.

1 Introduction

A corpus, as a collection of texts used as a representative sample of a given variety of a language or genre, is often considered as a tool in itself. Whether the investigator adopts a corpus-based approach, testing preformed hypotheses against authentic data, or a corpus-driven approach, inducing hypotheses from observed regularities or exceptions, corpora are an invaluable resource from which examples of *real language use* can be extracted to support linguistic arguments.

As soon as corpora could be stored electronically, tools were built to make the most of them. Over the years, corpus linguistics has thus equipped itself with numerous tools to meet various needs. Concordancers, for instance, are used to observe keywords in context relying on keyword-based queries. However using tools does not only allow corpus exploitation but also determines what observations can be made from them: what can be inferred from corpora strongly depends on the possibilities that the tool offers (Anthony, 2013), and relying on keywords alone are a drawback for those who are interested in complex constructions and/or constructions which do not have a specific lexical marker. We will consider the case of relative clauses, as they are not marked by one specific lexical item but by the whole grammatical category of relative pronouns.

It is hardly possible today to search for complex structures without knowing how they are analysed in the annotated corpus, which implies that one masters at least both a query language and the tagset of the annotated corpus. These skills are common in the fields of Computational Linguistics and Natural Language Processing (NLP) but require tremendous effort from non-specialists such as language learners or teachers to be grasped.

In this article, we will first account for the need of the use of native corpora in language learning and the tools currently available to explore them. We will then present a processing chain which is based on the notion of syntactic similarity and takes into consideration the potential difficulties encountered by non-specialists.

2 Corpus Query

Language learners and teachers are generally not linguists and are seldom familiar with methods from the Computational Linguistics or NLP fields despite their growing interest for corpora. Having explained the whys and wherefores of the access to authentic data in language learning, we will present current tools used to interrogate corpora as well as their limits, especially when the query focuses on a syntactic construction.

2.1 The use of corpora in language learning and teaching

Native corpora are interesting resources for language learning as they represent for both teachers and learners collections of authentic data in which it is possible to observe what is considered as natural or usual in the target language (see Chambers (2005; 2010) or Cavalla (2015) for examples of uses of corpora to improve writing skills in French as a foreign language). Exposure to authentic data can be indirect (for instance through the study of concordance print-outs carefully chosen by the teacher beforehand) or it can be the outcome of a more direct process. The latter is particularly exploited in what Johns calls *Data-Driven Learning*, which considers language learners as "research workers whose learning needs to be driven by access to linguistic data" (Johns, 1991). Learners should therefore be active in their learning process, being able not only to formulate hypotheses but also to observe and analyse linguistic data to confirm or refute their hypotheses by themselves, and eventually formulate new hypotheses if necessary.

However, in practice, learners might consider that the benefit gained from a direct confrontation with authentic corpora is not worth either the time or the effort put into learning how to use corpus exploration tools. Boulton (2012) conducted experiments involving his university students using corpora and mentioned that "causes of concern focused on the complexity of the interface (the functions and the query syntax) and the time it took to conduct some queries.". One of the students even expressed the need to attend a course specifically dedicated to the use of corpus exploration tools. Based on the same conclusions, Falaise et al. (2011) proposed an adapted exploration tool for treebanks displaying an interface that is simple, minimalist (options are hidden) and user-friendly (using a graphical interface rather than a textual one). This simplification does not hinder the expression of elaborate and precise queries but does not solve the problem either. Although users spend less time mastering this kind of tool, they still need to know how data are encoded in the tagged corpus.

2.2 Current methods for corpus query

One of the most common methods in Corpus Linguistics consists in using concordancers to look at language as it is. These tools are increasingly used in the context of language teaching and include at least two main functions : on the one hand, concordancers bring to light general statistical properties of a text or corpus (displaying lists of words with their frequencies, distributions, collocations etc.) and on the other hand, they also allow more detailed analysis with KWIC (KeyWord In Context) concordances, showing the target word or sequence of words aligned in their original context. It should be pointed out that unlike queries used in search engines, the sequences of words given as input to a concordancer are generally n-grams, in other words, sequences of n strictly contiguous words with a fixed order. The implementation of *skipgrams*, or non-contiguous n-grams, in concordance tools is quite rare but can be found in tools that focus on the search for phraseological units such as ConcGram or Lexicoscope for French. Both systems take as inputs several words[1] called *pivots*, either directly input by the user or found through iterative associations. In the latter case, the tool takes a first pivot (or the first two for ConcGram) and searches for words with which it has the strongest co-occurrence rates; these words are then used in turn as pivots (up to four additional pivots) (Cheng et al., 2006; Kraif and Diwersy, 2012).

In all the above-mentioned cases, queries are based on words. However, it is possible to go beyond words by resorting to morphosyntactic tags directly. The matching of two segments such as "*the person who is sleeping*" and "*the jury which was*

[1] By *words* we are referring to inflected forms of a word, but also to the corresponding lemma. It is therefore up to the user to choose whether morphological variations should be considered or not.

locked up" which have no lexical units in common but which share the same syntactic structure can only be achieved with a pattern like "`DET NOUN WH-PRO AUX VERB`"[2]. This type of query is commonly used in linguistics, but producing such patterns requires users not only to know the tagset of the corpus but also, and maybe more importantly, to be able to associate a word with the right part-of-speech. Regular expressions are a good means to broaden the range of query possibilities but at the cost of more advanced learning to attain that level of abstraction. GrETEL, a tool developed by Augustinus et al. (2012), partly solves the problem as it offers the possibility to interrogate a treebank by automatically transforming an example of a syntactic structure into a query, in the same manner as our proposal. This process is designed to spare users from learning a complex query syntax, but is still aimed at linguists who know what they are looking for and are capable of configuring the query to fulfill their purpose.

While our methodology relies on the same idea as GrETEL, we wish to go one step further in opening corpus exploration tools to a broader public. With this aim in view, our processing chain must (1) reduce the complexity of the interface of the query system and (2) reduce the depth and variety of knowledge required from the user. Incidentally, even though it might seem more relevant to work with treebanks, our research problem only focuses on the use of corpora annotated with morphosyntactic tags. We chose not to make use of dependency or constituency links yet for the sake of genericity, for treebanks are still rare resources.

3 Methodology

3.1 Processing chain

As our main objective is to simplify the query formulation as much as possible for non-specialists, we propose a methodology which takes as input a simple example of a target syntactic construction written in natural language and "directly" retrieves other examples of that construction. Every step from the transformation of the input into a query to the ranking of relevant sentences is performed by automatic processes and therefore does not require any more knowledge than that necessary to validate (or invalidate) the output.

The complete processing chain detailed in Figure 1 and illustrating a query on relative clauses with "who" is divided into six steps:

1. the input of one or several segments by the user[3] and expressed in natural language;
2. the conversion of the initial input into an actual machine-interpretable query using an automatic (morpho)syntactic analyser or parser;
3. the syntactic similarity measure between the query and sentences from the tagged corpus;
4. the proposition of relevant sentences grouped by clusters according to the mode of research;
5. the selection by the user of the example which seems to be the closest to his/her input or to what he/she expected to see, thus refining the initial query (selecting a relevant example narrows the number of matches and increases precision as retrieved segments must be similar to both the query and each newly appointed relevant example[4]);
6. the output of segments belonging to the chosen cluster (through the selection of its most representative example in the previous step).

As this project is still being developed, we focus on the first three steps of the process in this paper.

3.2 Similarity as a flexible search method

As we have seen with the example of relative clauses, syntactic similarity cannot rely on sequences of lexical units only but should rather be described with syntactic patterns in the form of sequences of syntactic tags possibly associated with

[2]These part-of-speech tags do not belong to any specific tagset. They are purposely generic and we decided to use the tag `VERB` for the sake of illustrating the fact that the two segments are different in terms of grammatical categories (auxiliary and -ing verb on the one hand, verb and preposition on the other hand) but are *similar* in the sense that they are both verbal phrases.

[3]Steps requiring an intervention by the user are represented by shapes with thick dark contours and were reduced to the strict minimum in compliance with our objective.

[4]Steps 4 and 5 are iterative, allowing the user to refine the query until satisfaction.

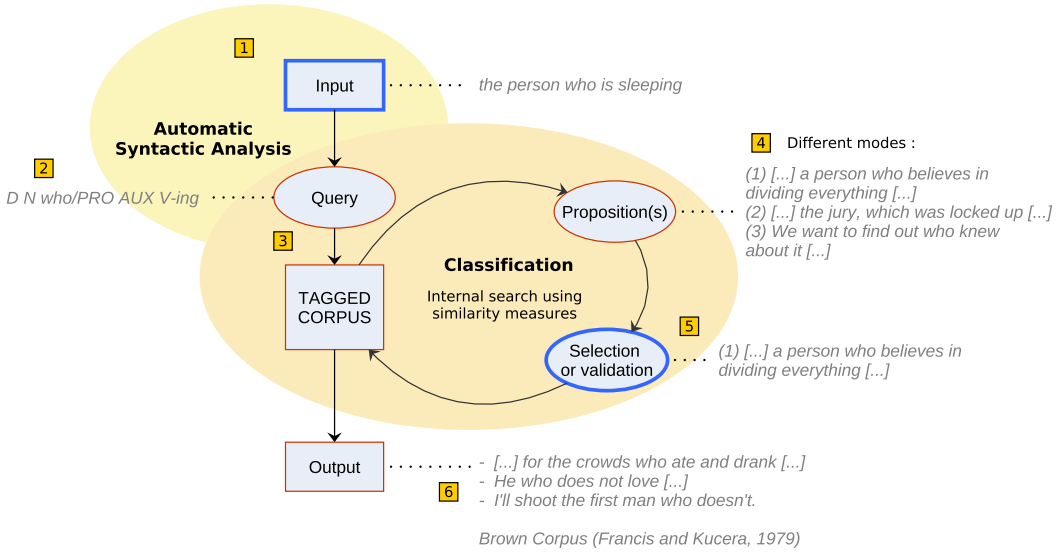

Figure 1: Flowchart of the process proposed for the syntactic query system

lexical units. The idea is to match instances of a syntactic construction while tolerating some variations in the vocabulary but also variations in the structure itself. Indeed, if we look at the propositions in step 4 of Figure 1, we notice that the first segment retrieved by the tool does not strictly match the query. The segment would be described by the sequence "`D N PRO V PREP V-ing`" while the query does not contain any preposition but the present progressive form of an intransitive verb. Despite these differences, this proposition is still relevant as it does display a relative clause and is similar enough to the initial query to be easily identified as such.

It is not self-evident for non-specialist users to define an efficient pattern, with a sufficiently high tolerance threshold to accept variations but low enough to keep a decent precision. We thus propose a method based on the similarity measure between an automatically defined pattern extracted from the input example(s) (the query) and examples from the corpus.

This methodology has a two advantages: it is more flexible than a query with regular expressions and it makes it possible to stay closer to data as it respects the bottom-up method supported by the *Data-Driven Learning* approach[5].

This flexibility also allows users to choose between different pre-defined options:

1. searching for similar segments with the same *grammatical* word(s), marked as relevant by the user (see the first proposition in step 4 of the flowchart);

2. searching for strongly similar segments but which do not contain the same *grammatical* word(s), marked as irrelevant by the user (see the second proposition, where "which" appears instead of "who");

3. searching for the same *grammatical* word(s), but in different contexts, in other words least similar sequences of tags (see proposition 3);

4. if lexical resources are available, searching for a semantically close *lexical* word could also be a possibility.

The second option could typically be used to search for structures such as relative clauses as they are characterised in English by the grammatical category of relative pronouns which have different surface forms among a finite list of possibilities. The

[5]As opposed to *data-based*, approaches called data-driven follow an inductive reasoning and start from the observation of regularities in data to formulate hypotheses or modify them.

tool must therefore be capable of identifying the category of relative pronouns but not necessarily try to match the one in the query, and more importantly, it must retrieve segments with variations in peripheral tags since the syntactic context could be quite different depending on the function of the pronoun (compare the constructions of *whom* and *whose* in "a few hundred people whom she knew" and "students whose interviews I discuss").

The first option gives outputs close to what can be retrieved through a concordancer, with the difference that the context also needs to be similar to that of the input query. As for the third option, it enables the user to search for other contexts of use of a specific word (or sequence of words), thus finding new functions for instance (see 4.3 for an example of an interesting case). Finally, option 4 would include the possibility to expand the query by using semantic similarity, as is commonly done in certain applications in information retrieval (search engines, question-answering systems) where keywords can be replaced by synonyms or hyperonyms.

Users can choose between these options from the beginning if they are sufficiently aware of what they are searching for and sufficiently competent to identify it. Otherwise, they can determine what suits them better from observing the concrete examples presented for each option (in the same manner as in step 4 and from comparing to what they expected.

3.3 Similarity measures

We chose to use Jaccard and Dice coefficients, widely used in NLP to measure similarity, in particular between two words or two strings. In our case, these coefficients can be used to compare larger units, such as sequences of syntactic tags (`D N PRO AUX V-ing`) or sequences containing tags associated with their lexical units (`N who/PRO AUX V-ing`). We are also exploring the possibilities offered by edit distance (or Levenshtein distance), a metric which is not an actual similarity measure but can evaluate indirectly the distance (dissimilarity) between two objects: if the similarity is maximal, the distance is zero. This alternative is particularly interesting as the edit distance between two "words" (or sequences of tags) M and N is defined by the minimal cost necessary to transform M into N through specific operations, the insertion, the deletion or the substitution of a unit (a character if it is a string or a tag in a sequence of tags for instance).

Even more interesting is the possibility to weight the cost of each operation and thus to adapt the distance to our data. With this method it would then be possible to consider the removal of an adjective (or all modifiers) as costing less than the removal of a verb or a conjunction for instance.

4 Preliminary experiments

We are currently conducting experiments on Korean as a foreign language, simulating queries that could be made by a learner of Korean who has difficulties apprehending a grammatical structure and understanding the contexts in which it is used (Wang, 2016).

4.1 Data

We considered that learners were likely to use our tool when failing to fully understand sentences they encountered. We thus decided to use as inputs for our preliminary experiments sentences that are typically available to learners, that is to say those used to illustrate grammatical points in grammar books or language textbooks. Accordingly, sentences extracted from textbooks of levels 1, 2 and 3 (equivalent to roughly three years of study of Korean) from Yonsei University and Ewha Language Center were gathered to make a corpus of potential inputs to our tool. The structure of the sentences was compared to those from the Sejong Corpus (Kim, 2007), the reference corpus for Korean language. Tests were made on the monolingual morphosyntactically annotated part of the Sejong Corpus (a total of around 13.5 million tokens) and composed of samples from various genres, including written essays to transcriptions of spontaneous conversations.

4.2 Method

There are two essential prerequisites to enable a syntactic comparison between an input query and sentences from a corpus: firstly, an efficient automatic morphosyntactic tagger or parser must be used on the input, and secondly, the tagset used by the tagger and the one that was applied on the corpus must be identical (in the case of similar tagsets, adaptations should be done beforehand). In our case, we

used an implemented version of KKMA[6], originally developed by the Intelligent Data System (IDS) Laboratory at Seoul National University and wrapped in KoNLPy[7] (Park and Cho, 2014). Among the five morphosyntactic analysers available in KoNLPy, KKMA was the slowest to run according to tests[8] but this flaw is not critical as KKMA would here be used to tag only one or a few sentences at most. Additionally, KKMA was trained on the Sejong Corpus, thus very few adaptations were needed to get perfectly matching part-of-speech tagsets.

내일은 맑**을지도 모릅니다.**
nay-il-un malk-**ul-ci-to mo-lup**-ni-ta.
'It is unsure if the weather is going to be clear tomorrow.'
↓
내일/NNG 은/JX 맑/VA **을지/EC 도/JX**
모르/VV ㅂ니다/EF ./SF

Figure 2: Example of input sentence

A typical input for our tool would be a sentence like the one in Figure 2: a sentence taken from Ewha's Korean Language textbook level 3-2, which has been segmented (essential for an agglutinative language like Korean) and annotated by KKMA. We set the morphemes illustrating the grammar point in bold. Likewise, users may eventually also have the possibility to show what morphemes seem to be their target, possibly in a simplified but similar manner as the matrix Augustinus et al. (2012) proposed for GrETEL, in which users are asked to choose if a word from the input is relevant or not and to what extent (relevant as POS, lemma or token). Sentences from Sejong Corpus are initially formatted the same way but would be output without tags so that users would really only see natural language sentences from input to output.

As for the technical aspect of our tool, different parameters were tested in our preliminary experiments:

- number of sentences as inputs: whether a single sentence was sufficient or if a greater number of sentences was more efficient;
- modes: whether the different options we described in this paper were relevant and viable;
- use of lexical units: whether we should include lexical units and take them into account in the similarity measure, or keep their part-of-speech tags only;
- similarity measure: whether a traditional similarity measure (such as Jaccard or Dice coefficients) is better than weighted edit distance or not;
- genres: whether all genres of texts or transcription types were relevant for our task, and which should be made default if any.

Current experiments focus on two differents types of grammar points of the Korean language which could be tricky to distant language learners: -(으)로 *-(u)lo*, the instrumental case particle which also fulfils different roles such as marking directions (학교**로** *hakkyo**lo*** 'in the direction of school') or the essive function (학생**으로** *haksayng**ulo*** 'as a student') and -(으)ㄹ지도 모르다 *-(u)lcito moluta*, a construction relying on several morphemes to express an epistemic modality (strong uncertainty).

4.3 Preliminary results

Results from our preliminary experiments are still too tentative to allow us to draw a clear conclusion on the most efficient parameters to represent input data or which measure should be applied and how.

However, we observed that:

- in most cases, one or two sentences given in input were sufficient to determine the context targeted. A greater number of sentences could be relevant if they all shared the same pattern, otherwise, it would only produce more confusion for the similarity measure;
- preliminary results from experiments on -(으)ㄹ지도 모르다 *-(u)lcito moluta* with the second option (same context, different

[6] http://kkma.snu.ac.kr/

[7] Korean Natural Language Processing in Python, an open source package supplying fundamental resources for Korean NLP. Experiments were run with KoNLPy 0.4.3.

[8] Time analysis and performance tests conducted by KoNLPy's development team are described on: http://konlpy.org/en/v0.4.4/morph/\#comparison-between-pos-tagging-classes

word(s)) confirm the ideas about why the different options could be theoretically interesting for the language learners we described in 3.2. Indeed, searching for a similar structure but a different morpheme retrieved sentences with -ㄴ/는지도 모르다 *-n/nuncito moluta* (see example (2a)), a structure absent from the textbooks we are working with despite a large number of occurrences in the Sejong Corpus and which is used to express a strong uncertainty as well, but without the prospective aspect of -(으)ㄹ지도 모르다 *-(u)lcito moluta*. This second option also retrieves allomorphs and other close constructions, respectively observed in examples (2b) and (2c). In contrast, sentences such as (1) retrieved using the first option (same context, same word(s)) simply contain the exact same construction as the one given in the input[9], i.e -을지도 모르(다) *-ulcito molu(ta)*;

(1) "다이아몬드가 붙을지도 모르지."
*taiamontuka puth**ulcito molu**ci*

(2) a. 끝내 망가뜨리고 말는지도 모른다
*kkuthnay mangkattuliko mal**nuncito molu**nta*
b. 그럴지도 모른다
*kule**lcito molu**nta*
c. 기대만큼 될지는 모르겠지만
*kitaymankhum toy**lcinun molu**keyssciman*

– deleting all lexical units could prevent our tool from retrieving certain structures relying on a lexical word, typically, -(으)ㄹ지도 모르다 *-(u)lcito moluta* which uses the verb 모르다 *moluta* 'to ignore'. Examples (3a) and (3b) were both retrieved using the second option but this time without lexical units. Only the sequence of POS (EC JX VV) has to be taken into account, resulting in very different constructions from the input. In the case of -(으)로 *-(u)lo*, deleting verbs of movement such as 가다 *kata* 'to go' or 내려오다 *naylyeota* 'to come down' could prevent our tool from discriminating between the directional function of the particle, often associated with such verbs, and other functions;

(3) a. 하긴 그렇기도 하겠네요
hakin kulehkito hakeyssneyyo
(POS-tagged form: 하긴/MAJ 그렇/VA 기**/EC** 도**/JX** 하**/VV** 겠/EP 네요/EF ./SF)
b. 이제 와서야 깨닫는다
icey waseya kkaytatnunta
(이제/MAG 오/VV 아서**/EC** 야**/JX** 깨닫**/VV** 는다/EF ./SF)

– edit distance has the advantage of retrieving sentences with similar length to the query, in our case, relatively short sentences, more likely to be of similar complexity as well. Other than that, no similarity measure seems to work better than another for now, but the weighting of edit distance costs could be refined with further experiments;

– experiments were only conducted on written texts (i.e samples from books, journals and newspapers). As searches focus on syntactic similarity instead of lexical words, all genres appear to be potentially relevant for language learners and allowing a search through all genres could raise awareness of extralinguistic factors such as the fact that newspapers and journals tend to be factual and do not contain as many occurrences of -(으)ㄹ지도 모르다 *-(u)lcito moluta* as in books.

The performance of such a tool is difficult to evaluate in terms of information retrieval quantitative measures since each retrieved sentence shares some similarity with the input and could therefore be considered as relevant. If we choose to focus on the quality of the system and the relevancy of the output sentences for users, we should ensure that the processing chain is working efficiently, which can be jeopardised by errors such as wrong POS tags in the very first step of our proposal. In order to be less dependent on the performance of the tagger or the parser, future experiments will include a non-corrected version of the Sejong corpus. With this

[9] Sentences from examples (1) to (3) were all extracted from Sejong's journal samples and were all retrieved using the sentence from Figure 2 as input and Levenshtein as the similarity measure.

method, potential tagging errors on the input would also be present in the corpus and match, while corresponding correctly tagged sentences from the gold standard version would be used as outputs for ethical reasons.

5 Conclusion and perspectives

We have seen that at the core of our study lies the simplification of the access to rich resources such as annotated corpora for a non-specialist public. Although certain studies support the idea that the confrontation with authentic data is beneficial even at an early stage of the learning process (Holec, 1990; Boulton, 2009), the potential complexity of authentic data raises the question of learners' autonomy. This tool is designed to be used by university students as well as self-directed language learners but the guidance of a teacher might be crucial for beginners, especially as we chose to explore monolingual corpora only. This work focuses on the design of the tool but several extra options will be studied to tackle this problem, including the categorisation of each sample in terms of genre and readability degree, a color-coded grammar so that learners can easily distinguish and identify parts-of-speech (similar to what was proposed for FipsColor (Nebhi et al., 2010)) or even an integrated monolingual or multilingual dictionary so that unknown vocabulary does not add another layer of cognitive difficulty to the analysis of the output. These enhancements which operate both at the very beginning and at the end of the process are already implemented in numerous tools (not necessarily built for educational purposes).

A certain number of other treatments that we hope to present in the near future are considered, including steps 4 to 6 of our processing chain, notably the clustering of relevant sentences. This particular step is crucial in reducing the perceived complexity of corpus exploration as it allows the user to glance immediately at the *type* of output instead of being submerged by an overwhelming number of unsorted sentences (other than by alphabetical order of the preceding or following word). Each cluster would be represented by the example that seems to be the most representative of all members of the cluster (the *centroid*). We believe that this step could also discriminate the different uses of polysemous particles such as -(으)로 *-(u)lo* based on the dissimilarity of contexts (only examples from the same context would be in the same cluster).

We are not building a pedagogical tool in itself, but we believe that this program could in the end complement current pedagogical resources by offering an original focus on the grammatical constructions of the target language.

References

Anthony Laurence. 2013. A critical look at software tools in corpus linguistics. *Linguistic Research*, 30(2):141–161.

Augustinus L., Vandeghinste V. and Van Eynde F. 2012. Example-based treebank querying. *Proceedings of eighth international conference on Language Resources and Evaluation (LREC'2012)*, p. 3161–3167.

Boulton Alex. 2009. Testing the limits of data-driven learning : language proficiency and training. *ReCALL*, 21(1):37–54

Boulton Alex. 2012. Beyond concordancing : Multiple affordances of corpora in university language degrees. *Procedia-Social and Behavioral Sciences*, 34:33–38.

Cavalla Cristelle. 2015. Collocations transdisciplinaires : réflexion pour l'enseignement. *Le problème de l'emploi actif et/ou de connaissances passives des phrasèmes chez les apprenants de langues étrangères*, E.M.E & Intercommunication.

Chambers Angela. 2005. Integrating corpus consultation in language studies. *Language learning & technology*, 9(2):111–125.

Chambers Angela. 2010. L'apprentissage de l'écriture en langue seconde à l'aide d'un corpus spécialisé. *Revue française de linguistique appliquée*, XV, 9–20.

Cheng Winnie, Greaves Chris and Warren Martin. 2006. From n-gram to skipgram to concgram. *International journal of corpus linguistics*, 11(4):411–433.

Falaise Achille, Tutin Agnès and Kraif Olivier. 2011. Exploitation d'un corpus arboré pour non spécialistes par des requêtes guidées et des requêtes sémantiques. *Actes de la 18e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2011)*, Montpellier, France.

Holec Henri. 1990. Des documents authentiques, pour quoi faire. *Mélanges Crapel*, 20:65–74.

Johns Tim 1990. Should you be persuaded: Two samples of data-driven learning materials. *Classroom Concordancing: English Language Research Journal*, 4:1–16.

Kim Hung-Gyu, Kang Beom-Mo and Hong Jungha 2007. 21st Century Sejong Corpora (to be) Completed. *The Korean Language in America*, 12, 31–42.

Kraif Olivier and Diwersy Sascha 2012. Le Lexicoscope : un outil pour l'étude de profils combinatoires et l'extraction de constructions lexico-syntaxiques. *Actes de la 19e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2012)*, p. 399–406.

Nebhi Kamel, Goldman Jean-Philippe and Laenzlinger Christopher 2010. FipsColor : grammaire en couleur interactive pour l'apprentissage du français. *Actes de la 17e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2010)*, Montréal, Canada.

Park Eunjeong L. and Cho Sungzoon 2014. KoNLPy: Korean natural language processing in Python. *Proceedings of the 26th Annual Conference on Human & Cognitive Language Technology*, Chuncheon, Korea.

Wang Ilaine. 2016. A syntax-based query system adapted to language learning and teaching. *American Association for Corpus Linguistics (AACL) and Technology for Second Language Learning (TSLL) Conference*, Ames, USA : Iowa State University. Poster presentation.