


Visualising collocation for close writing

J. C. Roberts¹ , P. W. S. Butcher¹ , R. Lew² , G. Rees³ , N. Sharma⁴  and A. Frankenberg-Garcia³ 

¹Bangor University, UK

² Adam Mickiewicz University, Poland

³ University of Surrey, UK

⁴ The Open University, UK

Abstract

We present how we have developed a visualisation tool and text editor to display collocations for the purpose of close writing. Collocations are words that combine together in a natural way. Our design study approach brought together a collaboration of experts in lexicography, language learning, and visualisation, starting with low-fidelity prototypes before developing fuller functional systems. We studied the challenge of how to visualise collocations, such to help language learners write more effectively. We have co-created (i) an expert-curated dataset of over 30,000 collocations, (ii) developed a text-editor which performs word analysis, and recommends collocations, and (iii) created several in-situ visualisations linked to the editor, to help users visualise and lookup collocations, and view example sentences. Every stage of development has been evaluated with language learners and other potential users, which has positively improved its design and functionality.

CCS Concepts

• *Human-centered computing* → *Information visualization*; • *Applied computing* → *Arts and humanities*;

1. Introduction

Words that combine together in a natural way are said to collocate. Writers choose words carefully to express their ideas. While there are many digital resources to help, such as dictionaries, thesauruses, grammar checkers, etc. writing is still challenging. Writers who are not writing in their first language or those who have limited experience, find it particularly difficult to choose the right word; and may select a word which may mean something similar but is not naturally used in the specific context. Krishnamurthy [Kri87] describes collocation as lexical items that occur “with a greater frequency than the law of averages would lead you to expect”. Texts that are created using common collocations are more readable and understandable by the reader. Collocations occur naturally. Indeed, the visualisation domain has its own set of familiar collocations [RFGL*18]. For instance, visualisation authors write *information visualisation*, not **visual information*, write *bar chart*, in preference to *charted bar* or *bar plot*, write *pie chart* and not **pie plot* [RAMB*19].

We present a design study to create an editor to help people write, and see collocating words in-situ. Although spelling corrections, grammar hints, word meanings, etc. are becoming more integrated with writing editors, such integration is not universal, and collocation in particular is not included or visualised [FG12, FG18, FGLR*19]. Our emphasis on collocation is important: the use of common collocations increases comprehension and flow of reading, collocations are processed more quickly

than free associations [VS19] and mastering them is the key to producing natural-sounding written work.

This paper presents ongoing work on the ColloCaid project [FGRL*20]. We demonstrate our current implementation of our close writing tool integrating various collocation visualisations. We explain how we carefully selected a representative set of 30,000 collocations and illustrative examples (Section 3). For our design process we started by sketching ideas (following the Five Design-Sheet method, Section 4), developed several prototypes (Section 5), which we incrementally improved following feedback from user-evaluation. We evaluated our working prototypes with real users, and particularly gained feedback from students on language courses, and researchers at conferences. Finally, we discuss our results and future work (Section 6).

2. Background and Related Work

As people write documents, they write, read, re-write and incrementally improve the document. Writers need to communicate their ideas effectively [FG18] and choose words that express the right meaning [Zak17]. Writing is often a cognitively-demanding task, particularly academic writing. One of the challenges though is that writers may not be aware of the limitations of their own texts [FG99]. Corpus linguistics can help. Using corpus linguistic techniques, learners can create a corpus of texts or, like us, use a pre-built corpus — we use the Oxford Corpus of Academic English in developing our dataset — to lookup words, and learn best prac-

tices through reading examples of how words are typically used in context by other writers. This is known as Data-Driven Learning (DDL) [Joh91, FG14, BC17]. There are many online corpora that could be used for developing writing in English such as British National Corpus (BNC) and the Corpus of Contemporary American English (COCA). And, with advances in corpus linguistics SketchEngine [KBB*14], Wmatrix [Ray08], CQPweb [Har12] and AntConc [Ant18]), it is much easier to create personal corpora. But, these tools are still at least one step removed from the writing process. What is required is therefore a closer integration with such linguistic techniques and the writing process.

There is much opportunity to visualise text data. Researchers have visualised corpus data using many techniques including: tag clouds plots [AC08], discourse trees [ZCCB12], dependency diagrams [CLD11a], and parallel coordinate plots (PCP) [CLD11b], and have reviewed text visualisation [KK15, LWC*18], text streams [ŠB10] and patents [FHKM16], and Alharbi and Laramée [AL19] classify and analyse 14 survey papers [AL19]. These strategies are useful to holistically display textual information, but they do not focus on the activity of writing. We summarise all this prior work as either *distant reading* or *close reading* techniques. For instance researchers have focused on how documents change over time, how they are structured, and topics of flow of ideas within the whole document. However in all this work, there is little mention of *close writing*.

Our focus is to visualise text to help writers. We want conventional combinations of words to be displayed to the user to help them in their task. It should not be distracting, allowing them to focus on their writing task. Subsequently, it is important to place additional information as close to the words being written (in-situ) without distracting the user. Most dictionaries rely on the writer to move away from their task and open another tool or window, to accomplish the task. Consequently, at the start of the research we asked several questions: How can we display collocations to users? How can the collocations be integrated with writing? Which collocations do we need to deliver to the users? How do we integrate visualisations such that users can also use standard functions, such as editing, spell checking, dictionary lookups, etc. alongside collocation information?

To address these questions, we chose to use an Agile design methodology, and brought together a diverse team of researchers skilled in linguistics, lexicography, writing pedagogy, human computer interaction (HCI) and visualisation. We chose this methodology because we had previous success with other projects, and it allows us to create usable software driven by user input. We had two main requirements: (1) we wanted to develop a tool where visualisations of word-collocations were integrated with a text editor. Users do not want to open additional windows, to search for a ‘language doubt’ [Has94] that may interrupt their writing flow. Learning in-situ is therefore useful, with language-suggestions provided in-context. The principal requirement is to provide suggestions which are triggered when a user types, which are defined from a carefully curated linguistic database. (2) We wanted a tool that looked familiar to the users. Our principal motivation is to help writers of English for Academic Purposes (EAP). Expert users who know the subject well are better at finding the words they require.

	AVL	AKL	ACL	Evidenced in all lists	At least two lists
Nouns	173	355	526	125	284
Verbs	130	233	96	38	136
Adjectives	86	180	83	24	94
Total	389	768	705	187	514

Table 1: Showing quantity of lemmas considered from English for our Academic Purposes corpus.

Base	Collocate	Score	Association	Example
equal	roughly	108	11.42	the latter two groups had roughly equal rates of
equal	nearly	57	9.89	3 experiments were performed using nearly equal
equal	relatively	20	5.48	all household members have relatively equal access
important	equally	401	10.35	equally important were the localisation of...
important	critically	124	9.13	determine which points of critically important info.
important	very	1495	10.03	It is very important for an economy to be stable

Table 2: Six examples from our collocation database, showing base, collocate and examples.

They find writing intuitive, and naturally choose familiar collocations and words that have the right meaning [Zak17]. However, non-expert writers may choose words that sound odd, and struggle to select suitable words to explain their concepts.

3. Curated data collection

The underlying collocational data are based on a carefully curated set of collocational bases (nouns, verbs, adjectives and prepositions), see Table 1. These bases are characteristic of Academic English, and we provide systematic coverage of over 30 thousand co-occurring words which are frequently used with them in academic texts. We derived the core lexical bases from analysing three vocabulary lists, extracting words that are found in (at least) two lists, and expanded this number by including homographs, e.g., we added content (verb) to complement content (noun), to arrive at 560 collocational bases. It would not be feasible to cover every possible collocation in a language. We aim to specifically to help writers with the collocations of academic English. English plays a fundamental role in the dissemination of knowledge, and focusing on academic English will enable us to develop a writing tool for a well-defined group of real-world users [FGLR*19]. It makes sense to prioritise more frequent words, since the words in a language tend to follow a Zipfian distribution [FG20].

The first list was a 389-item (excluding adverbs e.g., *however*, *therefore*) sub-list [Dur16] of the 3000-item Academic Vocabulary List (AVL) [GD14]. These 389 items are frequently present in student writing in 90% disciplines [AN09] found in the British Academic Written English corpus (BAWE) [Nes11]. This gave us a suitable candidate set of base words which academic writers were likely to use. The second list, the Academic Keyword List (AKL) [Paq10], was compiled by extracting keywords from the expert British EAP corpora and the LOCNESS corpus [Gra98] of British and American student written assignments. The third list came from 526 noun bases, 96 verb bases and 83 adjective bases, of the Academic Collocation List (ACL) [AC13]). We used SketchEngine’s [KBB*14] Word Sketch tool to identify the salient collocates found for these bases in expert academic writing, using the Oxford Corpus of Academic English. The thresholds used to select collocates were set after consultation with EAP experts. We selected collocations with frequency of ≥ 10 and logDice (associ-

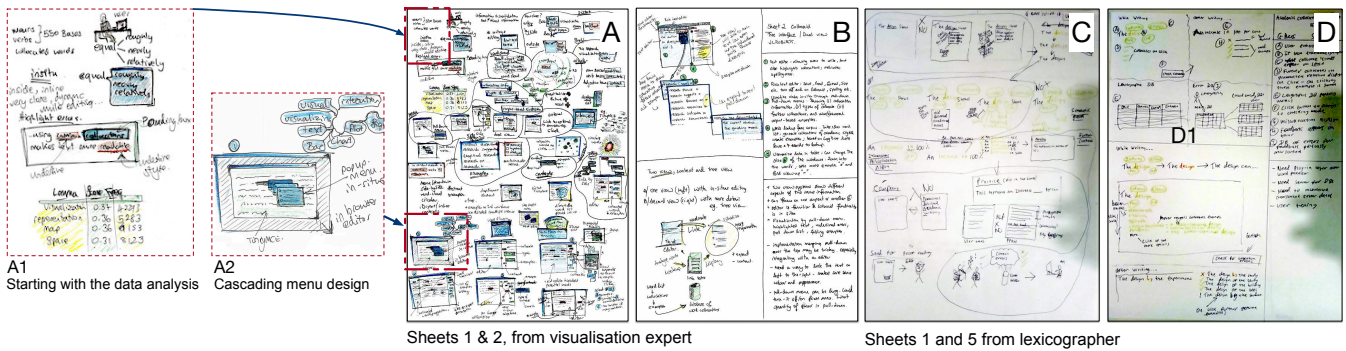


Figure 1: Selection of design sheets: A and B from the visualisation expert, showing how they started with the data analysis (A1), sketching many ideas including the cascading menus (A2). C and D from a lexicographer; showing several ideas including a tabular layout (D1).

ation) score of ≥ 5 for the parts of speech covered by our data set. Finally, we extracted three different examples for each collocation, from genuine academic texts, to show how they are used in context. Research indicates that three is more effective in encoding task than one example [FG12]. Table 2 shows some entries in our database for *equal* and *important*.

4. Data analysis, design sketches and low-fidelity prototypes

We started the design process by carefully considering data. Our primary data, shown in Table 2, includes the *base* word, the many associated collocates along with their occurrence score, and logDice [KBB*14] collocation association score, three examples per collocation. Consequently, we have quantitative data (co-occurrence frequency and association score), which can be used to arrange the collocations in order of their commonality. We have associative data, where the lemma is used in context with the collocating word(s). Readers will often understand collocations as a single block; therefore, we can treat the visualisation in a similar way, or as two independent words. In addition, we store the parts of speech: the verb, noun, adjectives and as dependent data. Collocations pattern into various syntactic types that are formed when combining words in respective syntactic classes, including: adverb+adjective, adjective+noun, noun+verb, verb+noun, verb-expression with preposition, verb+adverb.

Apart from our primary data, we can also visualise secondary data such as spelling errors, grammar mistakes, and punctuation, etc. Our dataset stores lemmas as headwords, but we can expand them into each word forms. For example, the lemma *run* (v) expands to *run*, *runs*, *running* and *ran*. We apply automated rules to expand the lemmas into the full lexical set (e.g., *deal* → *deal*, *deals*, *dealing*, *dealt*). We also apply rules to allow for spelling variants (e.g., *color* and *colour*, *ize* and *ise*). While this gives us a smaller lookup table and less storage, we lemmatise at run-time. Additionally, the writer could format the text, giving structure to the work with titles, sections numbers, figures, captions, etc. Potentially, each of this structured information could be visualised; however our focus is to help authors write better Academic English, therefore we focus on the written text and are less concerned with its structure or appearance.

Using the Five Design-Sheet (FdS) method [RHR16, RHR17],

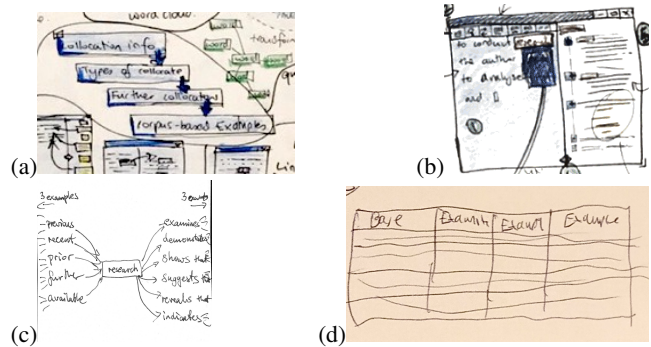


Figure 2: Several design ideas were common across designs. (a) drop-down in-situ visualisation, (b) dual view layout, (c) tree view, and (d) tabular layout showing words from the editor window.

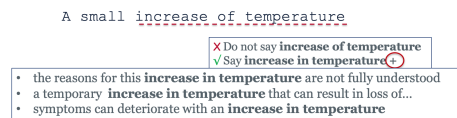


Figure 3: Wireframe mockups in Powerpoint™ to allow early low-fidelity demonstrations to occur.

we sketched different ideas. Sheet 1 allows the developer to explore many ideas, Sheets 2, 3 and 4 represent three different potential design solutions, and Sheet 5 the final design. Figure 1 shows two sheets from the visualisation expert, and two sheets from a lexicographer. These sheets were sketched independently. By performing this design process in parallel we can identify common ideas. In particular we noticed four common ideas, as shown in Figure 2: (a) a drop-down menu idea with collocation examples and methods to display whether the collocate is a verb, noun, adjective, etc.; (b) dual view system with in-line visualisations in the text editor and additional information in the side view; (c) a tree viewer showing the different collocations and three examples; and (d) a tabular layout of the words from the current editor view.

5. Prototype development

We first crafted a wire-frame mockup that we built in Powerpoint™ (Figure 3), which enabled us to demonstrate the principles at sev-

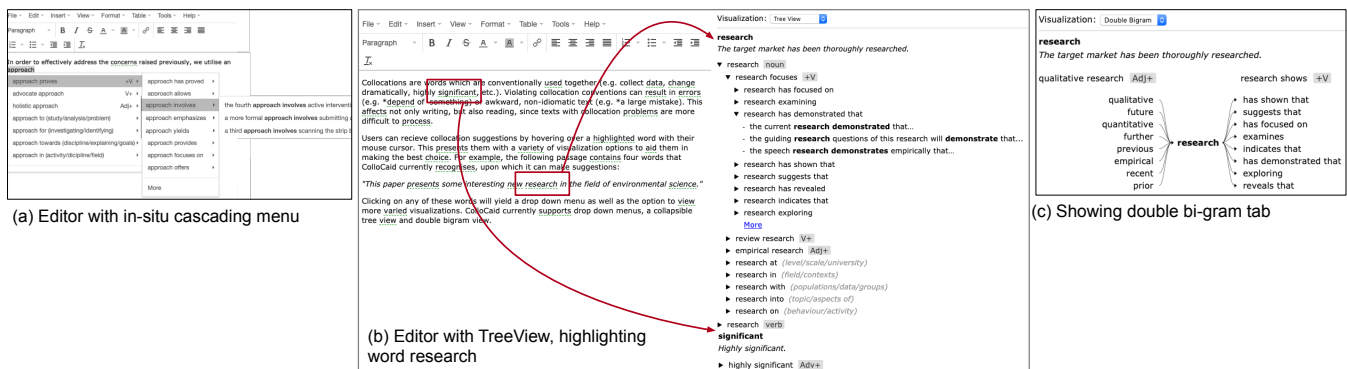


Figure 4: Screenshots from ColloCaid: (a) drop-down menus, and several examples for the word ‘approach’. (b) Tree view, where the user can then explore different collocations and examples (selected collocates for ‘research’ and ‘significant’ are shown). (c) Double bi-gram viewer, depicting strong collocates before and after the selected word.

eral workshops. This was important, because it allowed us to gain initial feedback. In particular, one user said “I’d really like to have the tool now, it looks like it would help me write better”.

To develop the prototype we knew that we wanted to have a familiar interface, but had to make a decision to use an open-source editor, or create the code from scratch. We decided on the former route, and took the decision to use TinyMCE, which is a fully functional editor, has a familiar interface, and is extendable. Figure 4a shows the main editor. Words get highlighted as the user types, demonstrating that there is collocation information. On display, we retrieve the strongest collocates using the logDice score, as stored in our data, and display them in the cascading menu in rank order (strongest collocates first) and organised using +V, +N, Adj+ etc. Finally we show three example sentences, ordered by their logDice score. We also dynamically visualise words as they are written in a dynamic tabular word viewer (not shown) that shows the last ten words typed with possible collocates and examples.

We evaluated this prototype (with only a fifth of the final database size) with nine participants in a workshop in Surrey, using think-aloud evaluation. We coded their suggestions in four ways: coverage, design, features and interaction of the tool. (i) With the smaller word set, it was not surprising that they wanted “more words to be highlighted”; (ii) they liked the design but suggested that more detail on the collocates could be displayed; (iii) they suggested extra features such as spelling, grammar editors etc.; (iv) they liked the interaction and the cascading interface, but a few participants did not like the dynamic tabular word viewer, one said the words “danced”.

From their critical feedback we improved the tool, adding in more data, and removing the tabular view. We evaluated this version with 141 participants across five sites: Leon, Paris, Porto Alegre, Sao Jose do Rio Preto, and Poznan. Participants tested the tool while engaged in authentic academic writing tasks. We asked participants to complete the System Usability Survey (SUS) [BKM08] with two additional questions “what did you like about the system” and “what could be improved in ColloCaid?” and asked participants to write open-ended comments. One user said “I like the fact that suggestions are immediately available without leaving the ed-

itor”, another wrote “I enjoyed the non-intrusiveness of the tool; I could look for more information if I wanted to”, and a third said “very user-friendly, it reminded me of a lot of collocates I had forgotten I knew”. We calculate the SUS results at the five sites to be: 84.2, 76.8, 78.5, 80.2, 79.9, which are encouraging scores, and means that participants view the tool to be “good” to “excellent” [BKM08].

We added a ‘more’ button on the words; a URL that redirects to SKELL (Sketch Engine for Language Learning) [KBB* 14] to provide additional information and examples about that word. We migrated to a dual-view system [Rob], with two further visualisations, Figure 4. The tree viewer (Figure 4b) shows more detailed information on a selected word. Users can unfold/fold the information to show more or less data. The collocation viewer (Figure 4c) depicts collocates to the right or left of the selected base word, which allows users to see, at a glance, different possible collocates.

6. Discussion and future work

We have developed a collocation editor, database and infrastructure that automatically looks up collocates for selected words, and visualises the collocation data in a tree, table and left/right collocate visualisations. Moreover, users can double click on the text in any window to automatically paste that word, collocation, example etc. into the editor. We started by creating early prototypes and performing ongoing and critical user feedback. Our user-evaluation demonstrates through high SUS scores that our participants view the tool to be usable. We have recently moved our tool behind an online login, so that we can discover who is using the tool and for how long. We currently have over 220 online users registered in our online trial. We have several additional evaluation sessions planned, and there are still improvements to make. For instance, the tree-view visualisations are coordinated one way: from the editor to the result. We have a demonstrator that synchronises the data both ways, which allows users to explore different collocates.

Acknowledgements

This research was supported by the Arts and Humanities Research Council, grant number AH/P003508/1.

References

- [AC08] ABBASI A., CHEN H.: CyberGate: A Design Framework and System for Text Analysis of Computer-Mediated Communication. *MIS Quarterly* 32, 4 (2008), 811–837. URL: <http://www.jstor.org/stable/25148873>.
- [AC13] ACKERMANN K., CHEN Y.-H.: Developing the Academic Collocation List (ACL)—A corpus-driven and expert-judged approach. *Journal of English for Academic Purposes* 12, 4 (2013), 235–247.
- [AL19] ALHARBI M., LARAMEE R.: SoS TextVis: An Extended Survey of Surveys on Text Visualization. *Computers* 8 (02 2019), 17. doi: 10.3390/computers8010017.
- [AN09] ALSOP S., NESI H.: Issues in the development of the British Academic Written English (BAWE) corpus. *Corpora* 4, 1 (2009), 71–83. doi:10.3366/E1749503209000227.
- [Ant18] ANTHONY L.: *Visualisation in corpus-based discourse studies*. Taylor and Francis, 1 2018, pp. 197–224. doi:10.4324/9781315179346.
- [BC17] BOULTON A., COBB T.: Corpus use in language learning: A meta-analysis. *Language Learning* 67, 2 (2017), 348–393. doi:10.1111/lang.12224.
- [BKM08] BANGOR A., KORTUM P. T., MILLER J. T.: An empirical evaluation of the System Usability Scale. *Human Computer Interaction* 24, 6 (2008), 574–594. doi:10.1080/10447310802205776.
- [CLD11a] CULY C., LYDING V., DITTMANN H.: XLDD: Extended linguistic dependency diagrams. In *Conference on Information Visualisation (IV)* (2011), IEEE, pp. 164–169. doi:10.1109/IV.2011.42.
- [CLD11b] CULY C., LYDINGAND V., DITTMANN H.: Structured Parallel Coordinates: a visualization for analyzing structured language data. In *Proc. 3rd Int. Conference on Corpus Linguistics, CILC-11* (2011).
- [Dur16] DURRANT P.: To what extent is the Academic Vocabulary List relevant to university student writing? *English for Specific Purposes* 43 (2016), 49–61. doi:10.1016/j.esp.2016.01.004.
- [FG99] FRANKENBERG-GARCIA A.: Providing student writers with pre-text feedback. *ELT Journal* 53 (1999), 100–106. doi:10.1093/elt/53.2.100.
- [FG12] FRANKENBERG-GARCIA A.: Raising teachers' awareness of corpora. *Language Teaching* 45, 4 (2012), 475–489. doi:10.1017/S0261444810000480.
- [FG14] FRANKENBERG-GARCIA A.: The use of corpus examples for language comprehension and production. *ReCALL* 26, 2 (2014), 128–146. doi:10.1017/S0958344014000093.
- [FG18] FRANKENBERG-GARCIA A.: Investigating the collocations available to EAP writers. *Journal of English for Academic Purposes* 35 (2018), 93–104. doi:10.1016/j.jeap.2018.07.003.
- [FG20] FRANKENBERG-GARCIA A.: Combining user needs, lexicographic data and digital writing environments. *Language Teaching* 53, 1 (2020), 29–43. doi:10.1017/S0261444818000277.
- [FGLR*19] FRANKENBERG-GARCIA A., LEW R., ROBERTS J. C., REES G. P., SHARMA N.: Developing a writing assistant to help EAP writers with collocations in real time. *ReCALL* 31, 1 (2019), 23–39. doi:10.1017/S0958344018000150.
- [FGRL*20] FRANKENBERG-GARCIA A., ROBERTS J. C., LEW R., REES G., BUTCHER P. W. S., SHARMA N.: ColloCaid – Find the words you need. <http://collocaid.uk>, cited April 2020.
- [FHKM16] FEDERICO P., HEIMERL F., KOCH S., MIKSCH S.: A survey on visual approaches for analyzing scientific literature and patents. *TVCG* 23, 9 (2016), 2179–2198. doi:10.1109/TVCG.2016.2610422.
- [GD14] GARDNER D., DAVIES M.: A new academic vocabulary list. *Applied linguistics* 35, 3 (2014), 305–327. doi:10.1093/applin/amt015.
- [Gra98] GRANGER S.: The computer learner corpus: a versatile new source of data for SLA research. In *Studies in Language and Linguistics*, Granger S., (Ed.). Routledge, 1998, ch. 1, pp. 3–18. doi: 10.4324/9781315841342-1.
- [Har12] HARDIE A.: CQPweb – combining power, flexibility and usability in a corpus analysis tool. *International Journal of Corpus Linguistics* 17, 3 (2012), 380–409. doi:10.1075/ijcl.17.3.04har.
- [Has94] HASSELGREN A.: Lexical teddy bears and advanced learners: a study into the ways Norwegian students cope with English vocabulary. *Applied Linguistics* 4, 2 (1994), 237–258. doi:10.1111/j.1473-4192.1994.tb00065.x.
- [Joh91] JOHNS T.: Should you be persuaded: Two samples of data-driven learning materials. *English Language Research Journal* 4 (1991), 1–16.
- [KBB*14] KILGARRIFF A., BAISA V., BUŠTA J., JAKUBÍČEK M., KOVÁŘ V., MICHELFEIT J., RYCHLÝ P., SUCHOMEL V.: The sketch engine: ten years on. *Lexicography* (2014), 7–36. doi:10.1007/s40607-014-0009-9.
- [KK15] KUCHER K., KERREN A.: Text visualization techniques: Taxonomy, visual survey, and community insights. In *PacificVis* (2015), IEEE, pp. 117–121. doi:10.1109/PACIFICVIS.2015.7156366.
- [Kri87] KRISHNAMURTHY R.: The process of compilation. *Looking up: An account of the COBUILD project in lexical computing* (1987), 62–85.
- [LWC*18] LIU S., WANG X., COLLINS C., DOU W., OUYANG F., EL-ASSADY M., JIANG L., KEIM D. A.: Bridging text visualization and mining: A task-driven survey. *IEEE TVCG* 25, 7 (2018), 2482–2504. doi:10.1109/TVCG.2018.2834341.
- [Nes11] NESI H.: BAWE: An introduction to a new resource. *New trends in corpora and language learning* (2011), 213–228.
- [Paq10] PAQUOT M.: *Academic vocabulary in learner writing: From extraction to analysis*. Bloomsbury Publishing, 2010.
- [RAMB*19] ROBERTS J. C., AL-MANEEA H. M. A., BUTCHER P. W. S., LEW R., REES G., SHARMA N., FRANKENBERG-GARCIA A.: Multiple Views: different meanings and collocated words. *Computer Graphics Forum* (3 2019). doi:10.1111/cgf.13673.
- [Ray08] RAYSON P.: From key words to key semantic domains. *International Journal of Corpus Linguistics* 13 (2008), 519–549. doi: 10.1075/ijcl.13.4.06ray.
- [RFGL*18] ROBERTS J. C., FRANKENBERG-GARCIA A., LEW R., REES G., SHARMA N.: Visualisation Approaches for Corpus Linguistics: Towards Visual Integration of Data-Driven Learning. In *3rd Workshop on Visualization for the Digital Humanities, at IEEE VIS'18* (2018).
- [RHR16] ROBERTS J. C., HEADLEAND C., RITSOS P. D.: Sketching Designs Using the Five Design-Sheet Methodology. *TVCG* 22, 1 (2016), 419–428. doi:10.1109/TVCG.2015.2467271.
- [RHR17] ROBERTS J. C., HEADLEAND C. J., RITSOS P. D.: *Five Design-Sheets – Creative design and sketching in Computing and Visualization*. Springer, 2017. doi:10.1007/978-3-319-55627-7.
- [Rob] ROBERTS J. C.: State of the Art: Coordinated & Multiple Views in Exploratory Visualization. In *CMV2007*, Andrienko G., Roberts J. C., Weaver C., (Eds.), pp. 61–71. doi:10.1109/CMV.2007.20.
- [ŠB10] ŠILIC A., BAŠIĆ B. D.: Visualization of text streams: A survey. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems* (2010), Springer, pp. 31–43.
- [VS19] VILKAITĖ L., SCHMITT N.: Reading collocations in an L2: do collocation processing benefits extend to non-adjacent collocations? *Appl. Linguist.* 40, 2 (2019), 329–354. doi:10.1093/applin/amx030.
- [Zak17] ZAKHAROV V.: Evaluation and combining association measures for collocation extraction. In *Proc IMS-2017* (2017), ACM, pp. 125–134. doi:10.1145/3143699.3143717.
- [ZCCB12] ZHAO J., CHEVALIER F., COLLINS C., BALAKRISHNAN R.: Facilitating Discourse Analysis with Interactive Visualization. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (Dec 2012), 2639–2648. doi:10.1109/TVCG.2012.226.