AUTHOR'S ACCEPTED MANUSCRIPT

Original article: B. Jurish, "Diachronic Collocations, Genre, and DiaCollo." In R. J. Whitt (ed.), *Diachronic Corpora, Genre, and Language Change*, John Benjamins, Amsterdam, 2018. DOI 10.1075/scl.85.03jur

Diachronic Collocations, Genre, and DiaCollo — REVISED DRAFT —

Bryan Jurish
Berlin-Brandenburgische Akademie der Wissenschaften
jurish@bbaw.de

Abstract

This chapter presents the formal basis for diachronic collocation profiling as implemented in the open-source software tool "DiaCollo" and sketches some potential applications to multi-genre diachronic corpora. Explicitly developed for the efficient extraction, comparison, and interactive visualization of collocations from a diachronic text corpus, DiaCollo is suitable for processing collocation pairs whose association strength depends on extralinguistic features such as the date of occurrence or text genre. By tracking changes in a word's typical collocates over time, DiaCollo can help to provide a clearer picture of diachronic changes in the word's usage, especially those related to semantic shift or discourse environment. Use of the flexible DDC search engine[1] back-end allows user queries to make explicit reference to genre and other document-level metadata, thus allowing e.g. independent genre-local profiles or cross-genre comparisons. In addition to traditional static tabular display formats, a web-service plugin also offers a number of intuitive interactive online visualizations for diachronic profile data for immediate inspection.

1 Introduction

DiaCollo is an open-source software tool for automatic *collocation profiling* (Church and Hanks 1990; Evert 2005) in diachronic corpora such as the *Deutsches Textarchiv*[2] (Geyken 2013) or the Corpus of Historical American English[3] (Davies 2012) which allows users to choose the projected collocate attributes and the granularity of the diachronic axis on a per-query basis (Jurish 2015; Jurish et al. 2016). Unlike conventional collocation extractors such as DWDS Wortprofil (Didakowski and Geyken 2013) or Sketch Engine (Kilgarriff and Tugwell 2002), DiaCollo is suitable for extraction and analysis of diachronic collocation data, i.e. collocation pairs whose association strength depends on the date of their occurrence and/or other extralinguistic features such as author or genre. By tracking changes in a word's typical collocates over time or corpus subset and applying J. R. Firth's famous principle that "you shall know a word by the company it keeps" (Firth 1957), DiaCollo can help to provide a clearer picture of associated changes in the word's usage.

Developed in the context of the European Union CLARIN project[4] to aid historians in their analysis of the changes in discourse topics associated with selected terms as manifested by changes in those terms' context distributions, DiaCollo has been successfully applied to

[1] "DWDS/Dialing Concordance", http://sourceforge.net/projects/ddc-concordance
[2] http://www.deutschestextarchiv.de
[3] http://corpus.byu.edu/coha
[4] http://www.clarin.eu

both mid-sized and large corpus archives, including the *Deutsches Textarchiv* (1600–1900, ca. 3.2K documents, 197M tokens) and a large heterogeneous newspaper corpus[5] (1946–2015, ca. 11M documents, 4.4G tokens). A modular web-service plugin provides access to the corpus hits for any returned collocation pair whenever the DiaCollo instance is associated with an independent DDC search engine, allowing users to proceed from the abstract, "distant" summary of a collocation profile to a more detailed "close" reading of the relevant corpus content (Moretti 2013).

The remainder of this chapter is organized as follows: after a brief discussion of previous work on both synchronic and diachronic collocation profiling in Section 2, the formal basis for diachronic collocation profiling as implemented in DiaCollo is presented in Section 3. Section 4 demonstrates some of DiaCollo's capabilities for genre-sensitive profiling by means of two simple examples. Finally, Section 5 summarizes the contribution and its implications for future work.

2 Related Work

A great deal of previous work on collocation discovery with respect to synchronic corpora can be found in the literature, including early work by Church and Hanks (1990), the textbook presentation by Manning and Schütze (1999), and more recent discussions by Evert (2005, 2008) and Rychlý (2008). The techniques and association measures developed for collocation discovery in synchronic corpora are well understood and have gained a wide degree of acceptance in the corpus linguistic community. Traditional approaches treat the source corpus as a homogeneous whole however: no provision is made for changes to a word's collocation behavior over time or corpus subset. Since co-occurrence frequencies are traditionally collected only for pairs of (surface) strings, the potential influence of occurrence date or extra-linguistic environment is irrevocably lost.

A number of diachronic corpus studies have made use of traditional collocation measures and other distributional features to track semantic change over time. Typically, such approaches begin by manually partitioning the corpus into "epochs" or "slices" by document date and proceed by performing an independent synchronic collocation analysis of each epoch, the results of which are then manually collated for interpretation with respect to a specific research question. Baker et al. (2008) for example partition their diachronic corpus into 10 annual sub-corpora, Sagi et al. (2009) use 5 corpus epochs each covering roughly 100 years, Gulordava and Baroni (2011) focus on 2 distinct epochs of 1 decade each, and Kim et al. (2014) manually partition their data into 160 epochs of 1 year each. Scharloth et al. (2013) exploit document metadata to implicitly partition a weekly newspaper corpus into ca. 3400 micro-epochs of one issue each, using a post-processing phase to categorize target terms and estimate frequencies as moving averages over a fixed-width temporal window, and Kilgarriff et al. (2015) describe a system for neologism detection using collocation analysis in any diachronic corpus admitting at least 3 distinct epochs. The vast discrepancies in number and size of corpus epochs is not arbitrary however. On the contrary, Gabrielatos et al. (2012) argue that the selection of temporal granularity must be dependent on the research question, with due consideration given to corpus size and expected frequency of the target item(s).[6]

[5]The *Alle Zeitungen* ('all newspapers') corpus hosted by the *Digitales Wörterbuch der deutschen Sprache* project ('digital dictionary of the German language', DWDS), cf. Geyken et al. (2017).

[6]As an anonymous reviewer pointed out, Gries and Hilpert (2008) made essentially the same observation

This criterion was one of the fundamental design goals of DiaCollo.

In many cases, diachronic corpus studies employ distributional-semantic vector-similarity measures rather than traditional collocation profiles. Wang and McCallum (2006) introduced the "Topics over Time" variant of latent Dirichlet allocation, which models a continuous time variable jointly with document-level word co-occurrences as mediated by a finite set of opaque topics. Sagi et al. (2009) employ latent semantic analysis to create a compressed vector-space model with respect to co-occurrence with the 2000 most frequent content-bearing collocates, and Kim et al. (2014) induce a series of vector-space models using neural networks as described by Mikolov et al. (2013). Relying as they do on compressed vector-space models defined in terms of opaque "topics" or "latent" dimensions, these approaches can provide at best a coarse approximation of a word's collocation behavior. While such approximations can be useful, their success crucially depends on the choice of compile-time parameters such as number of topics, which severely limits their applicability for generic diachronic collocation discovery.

3 Implementation

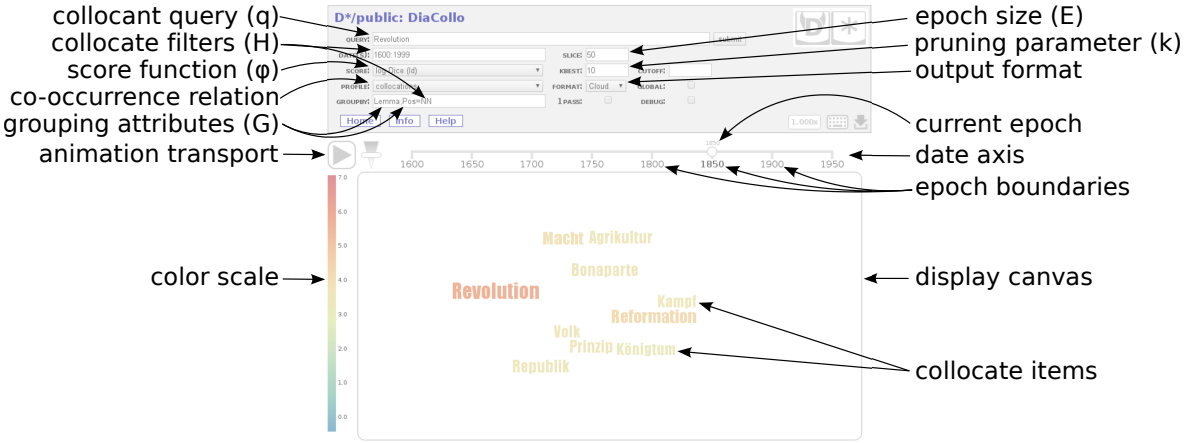
DiaCollo is implemented as a Perl library, and provides both a command-line interface as well as a modular RESTful web service plugin (Fielding 2000) with a form-based user interface for evaluation of runtime database queries and interactive visualization of query results. The remainder of this section presents the formal model of diachronic collocation profiles as implemented in DiaCollo. Section 3.1 provides a top-down sketch of DiaCollo's runtime functionality on the basis of a simple example using the form-based web interface. Section 3.2 describes the underlying corpus data model required for flexible attribute selection and on-the-fly diachronic partitioning. Acquisition of raw diachronic co-occurrence frequency distributions from suitably encoded corpus data is described in Section 3.3. Section 3.4 describes the online computation of association scores and the subsequent construction of a pruned diachronic profile from a raw co-occurrence frequency distribution, and Section 3.5 extends these techniques to provide direct diachronic comparisons of independent operand profiles.

3.1 Overview

Users typically interact with a DiaCollo corpus index through the web-service interface provided by the `DiaColloDB::WWW` module, an annotated screenshot of which is included here as Figure 1. In the simplest case, a user must provide at least a '`query`' parameter (q) indicating the collocant(s) for which a diachronic profile is to be computed; for the example in Figure 1, the selected collocant is the lemma *Revolution*. To acquire such a profile, the DiaCollo process must first identify all corpus lexemes matching the query and extract co-occurrence frequencies for each candidate collocate. In doing so, the corpus is implicitly partitioned into epochs of the size specified by the optional user parameter '`slice`' (E). Since not all lexical attributes may be relevant to a particular query, only those collocate attributes specified by

four years earlier, going on to propose a method for dynamically partitioning an input corpus into epochs of potentially non-uniform size. Their technique crucially relies on a pair of high-level functional parameters: a similarity measure and an amalgamation rule. Since instantiation of these parameters is in turn highly dependent on a particular user's research question, it is not immediately apparent how such a technique could be meaningfully implemented in the context of a user-agnostic diachronic profiling framework such as DiaCollo aims to provide.

Figure 1: Annotated screenshot of the web-based DiaCollo user interface displaying a dynamic tag-cloud visualization of the 10 best common noun collocates per 50-year epoch for the collocant *Revolution* over the interval 1600–1899. Variable names in parentheses are those used by the formal description in Sections 3.2–3.5.



the optional 'groupby' (G) parameter are projected onto the result set. The diachronic interval to be queried can be restricted by means of the 'date' parameter, and the admissible values of indexed lexical attributes can be specified by means of optional restriction clauses in the 'groupby' parameter (H). In Figure 1, collocates are grouped jointly by their 'Lemma' and 'Pos' (part-of-speech) attributes, the diachronic interval is restricted to 1600–1899, and only common noun collocates (Pos=NN) are considered.[7]

The precise meaning of "co-occurrence" depends on both the collocant query and the DiaCollo profiling relation specified by the 'profile' parameter (Section 3.3) – each co-occurrence relation supported by DiaCollo returns an epoch-labeled raw frequency profile for the requested collocant(s). Since raw frequency alone is often not a good indicator of association strength (Evert 2008), each candidate collocate is assigned a scalar association score by means of the quaternary operation specified by the 'score' (φ) parameter. Since the user is not typically interested in an exhaustive list of all candidate collocates, DiaCollo prunes the results to the highest-scoring candidates in each epoch, retaining only up to 'kbest' (k) collocates per epoch.

DiaCollo's web interface supports a number of different output formats for displaying diachronic profile data, specified by the 'format' parameter. Figure 1 shows a dynamic tag-cloud visualization using the D3.js library[8], which maps collocate items' association strength magnitude to font size and color according to a dynamically computed scale.[9] A horizontal diachronic axis acts as a slider, allowing users to "drag" a handle between individual epochs. For convenient inspection of diachronic development, this visualization mode also offers an animation transport in the form of a play/pause button and a playback speed slider: "playing" the animation causes the display canvas to interpolate smoothly between the discrete epochs represented by the underlying profile data, causing the display collocate items to change size

[7]The corpus in question uses the Stuttgart-Tübingen tagset (Schiller et al. 1995) to annotate part-of-speech tags. DiaCollo itself is language- and tagset-agnostic, but requires that all annotations to be indexed are present in the input corpus at index compilation time.

[8]http://d3js.org/

[9]Traditional static tabular display formats suitable for further processing are of course also available.

and color as their (interpolated) association scores change, and to fade in or out as they enter or leave the set of k-best collocates for the current epoch, respectively. Detailed information on collocate items themselves as well as hyperlinks to close approximations of the underlying corpus hits are available by a pop-up dialogue invoked by clicking on a item in the main display canvas.

The rest of this section will be concerned with the precise formal characterization of the corpus data model underlying DiaCollo's diachronic collocation profiles and of the compile- and run-time computations required for their acquisition. For further information on the web interface, its capabilities and limitations, and concrete usage examples, the interested reader is referred to the introductory tutorial[10] (in German) and the DiaCollo user documentation.[11]

3.2 Corpus Data

In order to track changes in a word's collocation behavior over time, we must first ensure that each corpus token is associated at least with the date of its occurrence. To this end, DiaCollo treats corpus tokens as n-tuples of potentially salient attribute values – including the occurrence date – rather than simple atomic strings. In the simplest case, a corpus need only provide surface strings $s_w \in \Sigma^*$ and associated dates $y_w \in \mathbb{N}$ as non-negative integers, encoding each token as a pair $w = \langle s_w, y_w \rangle$. Additional attributes may be encoded as well: lemmata are useful for abstracting over irregular inflection paradigms, part-of-speech tags and argument frames can help to isolate syntax-dependent phenomena, and document metadata such as author or genre can be used to restrict collocation profiles to a specific corpus subset of particular interest.

Formally, a corpus $\mathcal{C}$ is represented as a list of N tokens, each of which is represented as an n_A-tuple of attribute values drawn respectively from the sets $\mathcal{A}_1, \ldots, \mathcal{A}_{n_A}$, i.e. $\mathcal{C} = t_1 t_2 \ldots t_N \in (\mathcal{A}_1 \times \cdots \times \mathcal{A}_{n_A})^N$. Additionally, each corpus token t_i must be associated with the date of its occurrence, $\mathrm{Y}(t_i) \in \mathbb{N}$. The set of all types in the corpus modulo occurrence date is denoted by $\mathcal{W} = \bigcup_{i=1}^{N} \{t_i\} \subseteq \mathcal{A}_1 \times \cdots \times \mathcal{A}_{n_A}$, and $\mathcal{Y} = \bigcup_{i=1}^{N} \{\mathrm{Y}(t_i)\} \subset \mathbb{N}$ denotes the set of all occurrence dates in the corpus. I will also use the notation $t[j]$ to represent the projection of the jth attribute from the n-tuple t, $t[j] = a_j$ for $t = \langle a_1, \ldots, a_n \rangle$ and $1 \leq j \leq n$. By extension, $t[J] = \langle t[j_1], \ldots, t[j_{n_J}] \rangle$ denotes the projection of the attribute-list $J = \langle j_1, \ldots, j_{n_J} \rangle$ from t, and for a relation T of arity n (i.e. a set of n-tuples), $T[J] = \bigcup_{t \in T} \{t[J]\}$ denotes the projection of the attribute-list J from T. For an n_J-tuple $u \in T[J]$, $[u]_{T/J} \subseteq T$ denotes the equivalence class modulo J of u in T; $[u]_{T/J} = \{t \in T \mid t[J] = u\}$.

3.3 Co-occurrence Frequencies

Traditional collocation discovery methods for homogeneous synchronic corpora based on co-occurrence frequency distributions are well established and well understood. The most important question in the context of the current work is: how can we most effectively bring these methods to bear on heterogeneous diachronic data? Given an operational definition of what exactly it means for two words to "co-occur" in a corpus, conventional synchronic collocation profiling software such as DWDS Wortprofil (Didakowski and Geyken 2013) can compute all supported association scores for each collocation pair represented in the corpus offline, storing

[10] `http://kaskade.dwds.de/diacollo-tutorial`
[11] `http://kaskade.dwds.de/diacollo/help.perl`

these in a static database for efficient runtime retrieval.[12] For DiaCollo, such a static offline database is not feasible: not only should the user be provided with full runtime control of epoch granularity as argued by Gabrielatos et al. (2012), he or she should also be allowed to choose which of the indexed corpus attributes are to be projected onto the result set, and to restrict the profiled corpus subset by means of those attributes. Semantic preferences for example can be summarized well with a lemma-based collocation profile, disregarding differences in part-of-speech tag or surface form. Detection of genre-sensitive phenomena requires the ability to restrict the discovery procedure to one or more target genres, assuming these are appropriately encoded in the corpus data.

In place of a static association score database, DiaCollo offers several different runtime methods for acquiring a "raw" epoch-labeled absolute co-occurrence frequency profile from an arbitrary user request, which must then be split into independent epoch-wise sub-profiles, and the requested association scores computed on-the-fly from the raw frequency data for each collocate item in each sub-profile. While the computational load involved is substantially greater than that required for direct static database lookup, use of index structures optimized for sparse natural language data can provide sufficiently speedy access even for large source corpora.[13] The main advantage of the DiaCollo approach is the flexibility offered by user specification of epoch granularity, target collocant selection, and associated collocate grouping. DiaCollo's runtime computation of association scores from raw frequency profiles also makes it very easy to implement support for new association scores without the need for index re-compilation.[14]

Formally, a DiaCollo request is a 6-tuple $Q = \langle q, E, G, H, \varphi, k \rangle$, where:

- q is an expression selecting the target collocant(s) given by the request `query` parameter,
- $E \in \mathbb{N}$ is the size of the epochs into which the collocation profile is to be partitioned given by the request `slice` parameter,
- $G \in \langle g_1, g_2, \ldots, g_{n_G} \rangle$ is an n_G-tuple indicating the attributes to be projected onto the result as specified by the request `groupby` parameter,
- $H : \mathcal{Y} \times \mathcal{W}[G] \to \{0, 1\}$ is a Boolean-valued filter function determined by the request's `date` and `groupby` restriction clause parameters,
- $\varphi : \mathbb{R}^4 \to \mathbb{R}$ is an association score function given by the request `score` parameter, and
- $k \in \mathbb{N}$ is a non-negative integer specifying the maximum number of collocate items to return per epoch.

A raw frequency profile for a user request Q over epochs from the finite set $\mathcal{E} \subset \mathbb{N}$ is a 4-tuple $R_Q = \langle r_{\mathrm{N}}, r_1, r_2, r_{12} \rangle$, where:[15]

[12] In practice, some form of score- or frequency-based filter is usually applied in order to reduce storage requirements and improve access speed.

[13] An 8-epoch query over a 4.4 billion word corpus for example is evaluated using the native co-occurrence relation (Sec. 3.3.1) in under 5 seconds if the index data are present in the operating system's buffer cache, and in under 20 seconds if the data need to be paged in from disk.

[14] This feature could in principle be extended to support generic user-defined score function scripts in the spirit of *GraphColl* (Brezina et al. 2015).

[15] Note that the values of r_{N}, r_1, r_2, and r_{12} are *co-occurrence counts* and not necessarily traditional (unigram) frequencies. This is important for association score functions such as pointwise mutual information which properly operate on probabilities, in order to avoid overflow.

- $r_{\mathrm{N}} : \mathcal{E} \to \mathbb{N}$ maps each epoch to the total number of co-occurrences in that epoch,
- $r_1 : \mathcal{E} \to \mathbb{N}$ maps each epoch to the total independent frequency of the collocant(s) in that epoch,
- $r_2 : \mathcal{E} \times \mathcal{W}[G] \to \mathbb{N}$ maps each epoch-labeled collocate item to its independent frequency in the respective epoch, and
- $r_{12} : \mathcal{E} \times \mathcal{W}[G] \to \mathbb{N}$ maps epoch-labeled collocate items to the associated co-occurrence frequencies with the requested collocant(s).

The following subsections describe three of DiaCollo's runtime methods for raw frequency profile acquisition. These methods differ not only with respect to the underlying data structures employed, but also with respect to what exactly constitutes a corpus "co-occurrence" to be counted in the profile.

3.3.1 Native Co-occurrence Relation

For efficient profiling of co-occurrences within a fixed-width moving window of $\ell \in \mathbb{N}$ adjacent content tokens, DiaCollo uses a two-level native binary index I_{12} to associate pairs of fully specified attribute n_A-tuples with their absolute co-occurrence frequencies at each attested date unit. To populate I_{12} at index compilation time, the corpus is assumed to be partitioned into n_S contiguous segments[16] $s_1 s_2 \ldots s_{n_S} = \mathcal{C}$ with $s_i = s_{i1} s_{i2} \ldots s_{in_{s_i}}$ a list of n_{s_i} corpus tokens, and the native index is populated by counting co-occurrences in the specified window within each corpus segment as in (1):[17]

$$\begin{aligned} I_{12} \;&:\; \mathcal{W} \times \mathcal{W} \times \mathcal{Y} \to \mathbb{N} \\ &:\; \langle w, v, y \rangle \mapsto \sum_{i=1}^{n_S} \sum_{j=1}^{n_{s_i}} \sum_{j'=\max\{j-\ell,1\}}^{\min\{j+\ell,n_{s_i}\}} \mathbb{1}\left[j \neq j' \;\&\; s_{ij} = w \;\&\; s_{ij'} = v \;\&\; \mathrm{Y}(s_{ij}) = y \right] \end{aligned} \tag{1}$$

Independent occurrence frequencies are stored as true marginals over I_{12}:

$$I_1 : \mathcal{W} \times \mathcal{Y} \to \mathbb{N} : \langle w, y \rangle \mapsto \sum\nolimits_{v \in \mathcal{W}} I_{12}(w, v, y) \tag{2a}$$

$$I_{\mathrm{N}} : \mathcal{Y} \to \mathbb{N} : y \mapsto \sum\nolimits_{w \in \mathcal{W}} I_1(w, y) \tag{2b}$$

At runtime, a user collocant query q is first expanded to a set of attribute tuples $[\![q]\!] \subseteq \mathcal{W}$ – for a simple single-value single-attribute query such as "`$lemma=love`" this is simply a matter of selecting the appropriate tuples from the corpus vocabulary, $[\![\texttt{\$lemma=love}]\!] =$

[16] If available, sentence boundaries are good candidates for corpus segments. Implicit segment boundaries are inserted before and after each corpus source file. Since DiaCollo never counts co-occurrences crossing segment boundaries, this ensures that whenever a co-occurrence is counted, the co-occurring items do indeed share a common date label.

[17] $\mathbb{1}[\psi] \in \{0, 1\}$ is the indicator function for the truth-valued formula ψ; $\mathbb{1}[\psi] = 1$ if and only if ψ holds true for the current variable bindings, otherwise $\mathbb{1}[\psi] = 0$. Equation (1) thus counts exactly one co-occurrence of an ordered pair of terms $\langle w, v \rangle$ for each pair of distinct corresponding tokens ($w = s_{ij}$, $v = s_{ij'}$, $j \neq j'$) within a single corpus segment s which are separated by no more than ℓ intervening items ($-\ell \leq j' \leq \ell$). Note that identity pairs ($w = v$) will still be counted whenever multiple tokens of a single type co-occur within the selected context window.

$[\texttt{love}]_{\mathcal{W}/a_{\text{lemma}}}$. Then, a fully specified co-occurrence frequency distribution $\hat{I}_q$ at date-unit granularity can be computed as (3):

$$\hat{I}_q \quad : \quad \mathcal{Y} \times \mathcal{W} \to \mathbb{N} \; : \; \langle y, v \rangle \mapsto \sum\nolimits_{w \in [\![q]\!]} I_{12}(w, v, y) \tag{3}$$

Next, the co-occurrence distribution must be aggregated by the requested attributes G:

$$\hat{I}_{q,G} \quad : \quad \mathcal{Y} \times \mathcal{W}[G] \to \mathbb{N} \; : \; \langle y, g \rangle \mapsto \sum\nolimits_{v \in [g]_{\mathcal{W}/G}} \hat{I}_q(y, v) \tag{4}$$

The co-occurrence distribution must then be restricted to that subset of projected tuples satisfying the request filter function H as in (5):

$$\hat{I}_{q,G,H} = \hat{I}_{q,G} \restriction H^{-1}(1) \quad : \quad \mathcal{Y} \times \mathcal{W}[G] \to \mathbb{N} \; : \; \langle y, g \rangle \mapsto \begin{cases} \hat{I}_{q,G}(y, g) & \text{if } H(y, g) = 1 \\ \text{undefined} & \text{otherwise} \end{cases} \tag{5}$$

If $E > 0$ is the epoch size requested by the user via the `slice` option, then the function $\widetilde{E} : \mathcal{Y} \to \mathbb{N}$ maps each date value y to its associated epoch label $\widetilde{E}(y) = E\lfloor \frac{y}{E} \rfloor$.[18] DiaCollo epochs are thus labeled by their minimum possible element, so for a decade-wise partitioning ($E = 10$) the epoch label "1970" represents the interval 1970–1979. Let $\mathcal{E}_E \subset \mathbb{N}$ be the set of all zero-offset epochs of size E available in the corpus and let $[e]_E \subseteq \mathcal{Y}$ be the set of dates assigned to the epoch $e \in \mathcal{E}_E$; $\mathcal{E}_E = \widetilde{E}(\mathcal{Y}) = \bigcup_{y \in \mathcal{Y}} \{\widetilde{E}(y)\}$ and $[e]_E = \widetilde{E}^{-1}(e) = \{y \in \mathcal{Y} \mid \widetilde{E}(y) = e\}$. Then, the filtered distribution can be aggregated by epoch as in (6):

$$\hat{I}_{q,G,H,E} \quad : \quad \mathcal{E}_E \times \mathcal{W}[G] \to \mathbb{N} \; : \; \langle e, g \rangle \mapsto \sum\nolimits_{y \in [e]_E} \hat{I}_{q,G,H}(y, g) \tag{6}$$

The desired raw frequency co-occurrence distribution to be returned is then $\hat{I}_{q,G,H,E}$, and the independent profile frequencies r_1, r_2, and r_{N} can be acquired by consulting the marginal indices for each pair $\langle e, g \rangle \in \text{dom}(\hat{I}_{q,G,H,E})$:

$$r_{\text{N}}(e) = \sum\nolimits_{y \in [e]_E} I_{\text{N}}(y) \tag{7a}$$

$$r_1(e) = \sum\nolimits_{y \in [e]_E} \sum\nolimits_{w \in [\![q]\!]} I_1(w, y) \tag{7b}$$

$$r_2(e, g) = \sum\nolimits_{y \in [e]_E} \sum\nolimits_{v \in [g]_{\mathcal{W}/G}} I_1(v, y) \tag{7c}$$

$$r_{12}(e, g) = \hat{I}_{q,G,H,E}(e, g) \tag{7d}$$

Note that the runtime evaluation of Equations (3)–(7) can in most cases be efficiently performed with the help of appropriate auxiliary indices, without the need to iterate over all corpus tokens or types. Cached multi-maps for converting attested attribute values to the associated tuples in $\mathcal{W}$ for example enable efficient expansion of $[\![q]\!]$ and $[g]_{\mathcal{W}/G}$. Similarly, since only attested collocations with nonzero frequency[19] are stored in the index, evaluating Equation (3) will require iterating over only those candidate collocates which actually co-occur with the target collocant(s). Since these items are stored sequentially in the native index file, access speed for joint co-occurrence frequencies is improved even for large collocate

[18] The special case $E = 0$ is interpreted as a request for a synchronic profile over the entire corpus; $\widetilde{0}(y) = 0$ for all $y \in \mathcal{Y}$.

[19] Arbitrary attribute-wise and co-occurrence frequency threshold values may be specified at index compilation time to further compress the disk index and improve access speed.

sets on contemporary operating systems using a read-ahead cache. The filter function H and epoch partitioning $\widetilde{E}$ need only be evaluated for attested co-occurrences as well, and no explicit computation of the inverse relations H^{-1} and $\widetilde{E}^{-1}$ is required. In practice, the running time of native co-occurrence profiling is typically dominated by the retrieval of attested co-occurrences and their independent frequencies from the indices on disk as described by Equations (3) and (7c).

3.3.2 Term × Document Matrix Co-occurrence Relation

The fixed-width moving window notion of co-occurrence is not suitable for all applications. Assuming the default sentence segmentation granularity for example, very few potential collocates will be indexed for infrequent terms, leading to hyper-specific but uninterpretable result profiles for these items. In order to ameliorate such sparse data problems, most conventional distributional semantic models (Berry et al. 1995; Blei et al. 2003) represent the underlying corpus as a term × document frequency (TDF) matrix, which is then typically mapped to a low-dimensional approximation in terms of "latent" factors. In a similar spirit, DiaCollo offers a term × document co-occurrence relation drawing on a broader range of candidate collocates than those accessible by a short moving window. Unlike conventional distributional semantic models however, DiaCollo's TDF co-occurrence relation does not rely on opaque topics or latent factors to estimate term similarities, but provides exact co-occurrence counts for each attested collocation pair.

Formally, the TDF co-occurrence relation is defined in terms of a corpus partitioning into a set Doc of documents[20] and a frequency matrix $\mathrm{tdf} : \mathcal{W} \times \mathrm{Doc} \to \mathbb{N}$ such that $\mathrm{tdf}(w, d)$ is the frequency of the term w in document d. At index compilation time, DiaCollo explicitly creates and stores such a matrix together with an auxiliary vector mapping documents to the associated dates $\mathrm{dy} : \mathrm{Doc} \to \mathcal{Y}$, as well as an independent marginal date-frequency vector $\mathrm{yf} : \mathcal{Y} \to \mathbb{N} : y \mapsto \sum_{w \in \mathcal{W}} \sum_{d \in \mathrm{dy}^{-1}(y)} \mathrm{tdf}(w, d)$.

At runtime, a user request q is interpreted independently as a set of term-tuples $[\![q]\!]_{\mathcal{W}} \subseteq \mathcal{W}$ and a set of documents $[\![q]\!]_{\mathrm{Doc}} \subseteq \mathrm{Doc}$. If $[\![q]\!]_{/y} = [\![q]\!]_{\mathrm{Doc}} \cap \mathrm{dy}^{-1}(y)$ represents the subset of queried documents with date $y \in \mathcal{Y}$, then a TDF co-occurrence frequency distribution $\hat{I}_{\mathrm{tdf}:q,G}$ at date-unit granularity over the requested projection attributes G is computed as (8):

$$\begin{aligned} \hat{I}_{\mathrm{tdf}:q,G} &: \mathcal{Y} \times \mathcal{W}[G] \to \mathbb{N} \\ &: \langle y, g \rangle \mapsto \sum_{d \in [\![q]\!]_{/y}} \min \left\{ \Big(\sum_{w \in [\![q]\!]_{\mathcal{W}}} \mathrm{tdf}(w, d) \Big), \Big(\sum_{v \in [g]_{\mathcal{W}/G}} \mathrm{tdf}(v, d) \Big) \right\} \end{aligned} \tag{8}$$

Candidate filtering and epoch aggregation are analogous to the procedures described by Equations (5) and (6) for the native co-occurrence relation, and the final TDF raw frequency profile to be returned is defined for each pair $\langle e, g \rangle \in \mathrm{dom}(\hat{I}_{\mathrm{tdf}:q,G,H,E})$ as:

$$r_{\mathrm{N}}(e) = \sum_{y \in [e]_E} \mathrm{yf}(y) \tag{9a}$$

$$r_1(e) = \sum_{y \in [e]_E} \sum_{w \in [\![q]\!]_{\mathcal{W}}} \sum_{d \in [\![q]\!]_{/y}} \mathrm{tdf}(w, d) \tag{9b}$$

$$r_2(e, g) = \sum_{y \in [e]_E} \sum_{v \in [g]_{\mathcal{W}/G}} \sum_{d \in \mathrm{dy}^{-1}(y)} \mathrm{tdf}(v, d) \tag{9c}$$

$$r_{12}(e, g) = \hat{I}_{\mathrm{tdf}:q,G,H,E}(e, g) \tag{9d}$$

[20] By default, DiaCollo uses paragraph boundaries as "documents".

As for native profiling, the runtime evaluation of TDF co-occurrence profiles can be optimized by use of appropriate data storage formats and access methods. DiaCollo employs a pair of Harwell-Boeing offset pointers (Duff et al. 1989) to optimize access to the sparse TDF matrix stored on disk. Index lookup, aggregation, and filtering are performed by optimized routines written in the Perl Data Language "PDL" (Glazebrook and Economou 1997) for manipulation of large numerical data structures, using the operating system's memory-mapping facility for transparent on-demand paging.

3.3.3 DDC Co-occurrence Relation

DiaCollo's DDC co-occurrence relation makes use of the DDC search engine (Sokirko 2003; Jurish et al. 2014) to acquire raw frequency profiles from a running DDC server.[21] Unlike the native co-occurrence index, which implicitly defines a "co-occurrence" of two corpus items to be any occurrence of both items within a fixed-width moving window over the corpus, the DDC co-occurrence relation makes no implicit assumptions regarding which corpus configurations constitute "co-occurrences". Rather, DiaCollo's DDC back-end makes use of user-supplied search term subscripts ("match-IDs") to identify which elements of the query-string q represent the collocant(s) and which represent the associated collocates. By convention, collocant terms are identified by the subscript '`=1`' and the associated collocate terms are identified by the subscript '`=2`'. Query terms without an explicit subscript are treated as collocant restrictions and implicitly assigned the subscript '`=1`'.

By using subscripted DDC queries, the definition of what exactly constitutes a "co-occurrence" depends entirely on the query specified by the user, which may be any context query expressible in the DDC query language.[22] In particular, the DDC query language supports arbitrary corpus segmentations (e.g. sentences, paragraphs, files), Boolean expressions on both the token- and the segment-level, phrase- and proximity queries (e.g. immediate predecessor, immediate adjacency), bibliographic metadata filters and grouping, and server-side term expansion pipelines (e.g. thesauri).

This flexibility comes at a price however: since DDC itself was designed as a search engine rather than a collocation database, the entire corpus must be searched and each occurrence explicitly identified in order to acquire a raw co-occurrence frequency distribution. In particular, acquisition of the independent collocate-frequency distribution r_2 may need to process a huge number of individual occurrences, which makes DiaCollo's DDC back-end a comparatively slow and resource-hungry profile acquisition method.[23]

[21] A companion DDC index for the underlying corpus must be independently configured and created.

[22] `http://odo.dwds.de/~jurish/software/ddc/querydoc.html`

[23] Performance of the DDC back-end depends heavily on the user query and the size of the underlying corpus. Simulating the native co-occurrence relation's moving context window for example using the DDC query `NEAR(*=2,`q`,`$\ell-1$`)` first requires an inexpensive index lookup of $O(|q|\log(|\mathcal{W}|))$, followed by explicit processing of $f_{12} \approx 2\ell\mathrm{f}(q)$ co-occurrences for evaluation of (10d) and (10b). Heaps' (1978) law indicates that the number of candidate collocates discovered will be approximately $n_2 \approx K\sqrt{f_{12}}$ for some $K \geq 10$, and their total expected independent frequency can be estimated *ceteris paribus* as $f_2 \approx n_2 N/|\mathcal{W}|$, so the processing time for DDC frequency profile acquisition is dominated by (10c) with $O(n_2 \log(|\mathcal{W}|) + f_2)$. Since DDC indices in general store every token in the original corpus – including closed-class items – the candidate token count f_2 can in practice grow very large even for "simple" collocant queries with only a few co-occurrences f_{12}. Evaluating all of (10a)–(10d) using a DDC-based approximation of the native index query from Footnote 13 on the same 4.4 billion word corpus requires more than 5 minutes to complete even for an immediate adjacency query ($\ell = 1$), and causes DiaCollo to return an error message indicating that the underlying search engine query timed out. The epoch-size query of (10a) necessarily covers even more tokens than (10c), but since

Raw frequency profiles are generated for the DDC co-occurrence relation for a user request Q by means of four separate DDC query requests – one request for each component of the frequency profile:

$$r_{\mathrm{N}} = \lambda_q \times \texttt{COUNT(* \#SEP) \#BY[date/}E\texttt{]} \tag{10a}$$

$$r_1 = \lambda_q \times \texttt{COUNT(KEYS(}[\![q\&H]\!]\texttt{ \#SEP \#BY[}G\texttt{=1]) \#SEP) \#BY[date/}E\texttt{,}G\texttt{=1]} \tag{10b}$$

$$r_2 = \lambda_q \times \texttt{COUNT(KEYS(}[\![q\&H]\!]\texttt{ \#SEP \#BY[}G\texttt{=2]) \#SEP) \#BY[date/}E\texttt{,}G\texttt{=2]} \tag{10c}$$

$$r_{12} = \texttt{COUNT(}[\![q\&H]\!]\texttt{ \#SEP \#BY[date/}E\texttt{,}G\texttt{=2])} \tag{10d}$$

Here, $[\![q\&H]\!]$ is a DDC query string representing the logical conjunction of the request query q and filter conditions H. λ_q is a scaling coefficient heuristically determined by the query q which ensures that the joint and independent frequencies returned are compatible. A simple immediate-successor query (`"love *=2"`) for example will be assigned $\lambda_q = 1$, since at most one collocate occurrence can be identified for each occurrence of the collocant, while an immediate adjacency query such as `NEAR(love,*=2,0)` will be assigned $\lambda_q = 2$ since up to two co-occurrences (left + right) will be counted for each occurrence of the collocant.

3.4 Scoring and Pruning

All association score functions supported by DiaCollo are defined in terms of the frequency values provided by a raw co-occurrence profile $R_Q = \langle r_{\mathrm{N}}, r_1, r_2, r_{12}\rangle$ for the user request Q. The default score function used by DiaCollo is the scaled log-Dice ratio as introduced by Rychlý (2008). For details on the relative merits of other supported score functions, the interested reader is referred to the excellent discussion by Evert (2008). Formally, an association score function is a quaternary operation on real numbers $\varphi : \mathbb{R}^4 \to \mathbb{R}$. Independent score profiles $p_{Q,e}$ are computed for each epoch $e \in \mathcal{E}_E$ by applying φ to each candidate collocate item in turn:

$$p_{Q,e} \;:\; \mathcal{W}[G] \to \mathbb{R} \;:\; g \mapsto \varphi(r_{\mathrm{N}}(e), r_1(e), r_2(e,g), r_{12}(e,g)) \tag{11}$$

Since the user is typically not interested in an exhaustive profile of all potential collocates, DiaCollo prunes each epoch-local profile $p_{Q,e}$ to its k-best collocates before returning the results.[24] For $k \in \mathbb{N}$ and a real-valued function f, let $\mathrm{best}_k(f) \subseteq \mathrm{dom}(f)$ represent the set of k elements from $\mathrm{dom}(f)$ with maximal values under f; $|\mathrm{best}_k(f)| = \min\{k, |\mathrm{dom}(f)|\}$ and $f(x) \geq f(y)$ for all $x \in \mathrm{best}_k(f)$ and all $y \in \mathrm{dom}(f)\backslash\mathrm{best}_k(f)$. The epoch-local profile $\hat{p}_{Q,e} \subseteq p_{Q,e}$ resulting from k-best pruning can then be expressed as (12):

$$\hat{p}_{Q,e} \;=\; p_{Q,e}{\restriction}\mathrm{best}_k(p_{Q,e}) \;:\; g \mapsto \begin{cases} p_{Q,e}(g) & \text{if } g \in \mathrm{best}_k(p_{Q,e}) \\ \text{undefined} & \text{otherwise} \end{cases} \tag{12}$$

The final diachronic profile $\hat{P}_Q$ returned by a unary DiaCollo request Q is then simply a function mapping epoch labels to the corresponding pruned sub-profiles:

$$\hat{P}_Q \;:\; \mathcal{E}_E \to \mathbb{R}^{\mathcal{W}[G]} \;:\; e \mapsto \hat{p}_{Q,e} \tag{13}$$

its `#BY`-clause does not refer to any token-level attributes, it can be evaluated by an optimized subroutine in near-constant time, typically under 100 milliseconds. Since the native and TDF relations do not store or process individual tokens, the performance of (7c) and (9c) is limited for analogous queries only by the number of candidate collocate types to $O(n_2 \log(|\mathcal{W}|))$.

[24] An alternative pruning method allows the user to specify a minimum score threshold for collocate items.

Table 1: Comparison operations supported by DiaCollo in "diff" mode. "Pre-trimmed" comparison operations are defined over the union of pruned operand domains $\mathrm{Dom}_{Q_a \ominus Q_b / e_{ab}} = \mathrm{dom}(\hat{p}_{Q_a,e_a}) \cup \mathrm{dom}(\hat{p}_{Q_b,e_b})$, while "restricted" operations use the intersection of the un-pruned operand domains, $\mathrm{Dom}_{Q_a \ominus Q_b / e_{ab}} = \mathrm{dom}(p_{Q_a,e_a}) \cap \mathrm{dom}(p_{Q_b,e_b})$.

Label	**Domain**	$x \ominus y$	**Description**
`diff`	pre-trimmed	$x - y$	raw difference
`adiff`	pre-trimmed	$\lvert x - y \rvert$	absolute difference
`max`	restricted	$\max\{x, y\}$	maximum
`min`	restricted	$\min\{x, y\}$	minimum
`avg`	restricted	$\frac{1}{2}(x + y)$	arithmetic average
`havg`	restricted	$\frac{1}{2}\left(\frac{2xy}{x+y} + \frac{1}{2}(x + y)\right)$	pseudo-harmonic average

3.5 Comparisons

In addition to simple requests which return a score profile for the specified collocant(s), DiaCollo also offers a "comparison" or "diff" mode, by means of which the user may request a summary of the most prominent similarities or differences between two independently evaluated queries, e.g. between two different words or between occurrences of the same word in different date intervals, different text genres, or in the works of different authors. The first step in computing a diachronic comparison profile for two independent query requests Q_a and Q_b is to define an epoch-alignment $\mathcal{E}_{a \bowtie b} \subseteq \mathcal{E}_{E_a} \times \mathcal{E}_{E_b}$. DiaCollo supports both trivial alignments with singleton epoch domains as in (14a) and alignments of uniform-sized epoch domains as in (14b), where $X^{\leq}$ is the $|X|$-tuple resulting from sorting the elements of the finite set $X \subset \mathbb{N}$ by the natural order $\leq$.[25] Query pairs whose epoch domains do not satisfy either of these conditions cause DiaCollo to return an error message indicating that the alignment failed.

$$\mathcal{E}_{a \bowtie b} = \begin{cases} \mathcal{E}_{E_a} \times \mathcal{E}_{E_b} & \text{if } \min\{|\mathcal{E}_{E_a}|, |\mathcal{E}_{E_b}|\} \leq 1 \quad (14a) \\ \bigcup_{i=1}^{N_\mathcal{E}} \left\{ \langle \mathcal{E}_{E_a}^{\leq}[i], \mathcal{E}_{E_b}^{\leq}[i] \rangle \right\} & \text{if } |\mathcal{E}_{E_a}| = |\mathcal{E}_{E_b}| = N_\mathcal{E} \quad (14b) \\ \text{undefined} & \text{otherwise} \quad (14c) \end{cases}$$

Given such an epoch alignment $\mathcal{E}_{a \bowtie b}$ and epoch-wise operand profiles $\{p_{Q_a,e_a}\}_{e_a \in \mathcal{E}_{E_a}}$ and $\{p_{Q_b,e_b}\}_{e_b \in \mathcal{E}_{E_b}}$, a diachronic comparison sub-profile $p_{Q_a \ominus Q_b, e_{ab}}$ is computed for each pair of aligned epochs $e_{ab} = \langle e_a, e_b \rangle \in \mathcal{E}_{a \bowtie b}$ by applying a binary comparison operation $\ominus : \mathbb{R}^2 \to \mathbb{R}$ to each collocate item in turn:[26]

$$p_{Q_a \ominus Q_b, e_{ab}} \;:\; \mathrm{Dom}_{Q_a \ominus Q_b / e_{ab}} \to \mathbb{R} \;:\; g \mapsto p_{Q_a,e_a}(g) \ominus p_{Q_b,e_b}(g) \quad (15)$$

Here, $\mathrm{Dom}_{Q_a \ominus Q_b / e_{ab}} \subseteq \mathrm{dom}(p_{Q_a,e_a}) \cup \mathrm{dom}(p_{Q_b,e_b}) \subseteq \mathcal{W}[G]$ is the characteristic domain of the comparison profile, which depends on the comparison operation $\ominus$. A list of selected comparison operations supported by DiaCollo and their characteristic domains is given in Table

[25] In the simplest case, Q_a and Q_b share the same epoch domain $\mathcal{E} = \mathcal{E}_{E_a} = \mathcal{E}_{E_b}$, and the alignment $\mathcal{E}_{a \bowtie b}$ will be the identity relation over the shared domain, $\mathcal{E}_{a \bowtie b} = \mathrm{Id}(\mathcal{E}) = \{\langle e, e \rangle\}_{e \in \mathcal{E}}$.

[26] Undefined operand values are treated as zeroes when computing comparison scores via Equation (15). The DiaCollo API also ensures that the projected attributes and score functions of comparison operand requests are compatible, $G_a = G_b$ and $\varphi_a = \varphi_b$.

1. The raw difference operation $\ominus_{\texttt{diff}}$ selects collocate items which associate strongly only with Q_a, while the default comparison operation $\ominus_{\texttt{adiff}}$ selects those items which associate strongly with only one of Q_a or Q_b, regardless of which collocant is preferred. Collocates showing strong associations for both operand profiles can be selected with the $\ominus_{\texttt{havg}}$ operation, which uses the harmonic average of operand scores.[27]

Comparison profiles must then be pruned to their k-best collocates as given in Equation (16), analogous to the unary profile pruning procedure described in Section 3.4. In the case of comparison profiles however, the score values to be returned may be distinct from those used for k-best selection. In particular, the default absolute difference comparison operation $\ominus_{\texttt{adiff}}$ selects the k collocates with maximally dissimilar association scores between its operand profiles, regardless of which of the operands displays the stronger preference (as expressed by the sign of the raw difference score). For intuitive interpretation of comparison profile results, the returned values should retain the information provided by the sign of the raw difference score. To this end, each comparison operation $\ominus$ is implicitly associated with a companion operation $\boxminus$ for producing final return values. The absolute difference operation is configured to return raw difference scores by setting $\boxminus_{\texttt{adiff}} = \ominus_{\texttt{diff}}$. For all other comparison operations δ, $\boxminus_\delta = \ominus_\delta$ and Equation (16) is completely analogous to the unary profile pruning procedure given by Equation (12).

$$\begin{aligned} \hat{p}_{Q_a \ominus Q_b, e_{ab}} &= p_{Q_a \boxminus Q_b, e_{ab}} \restriction \text{best}_k(p_{Q_a \ominus Q_b, e_{ab}}) \\ &: g \mapsto \begin{cases} p_{Q_a \boxminus Q_b, e_{ab}}(g) & \text{if } g \in \text{best}_k(p_{Q_a \ominus Q_b, e_{ab}}) \\ \text{undefined} & \text{otherwise} \end{cases} \end{aligned} \tag{16}$$

The final diachronic comparison profile $\hat{P}_{Q_a \ominus Q_b}$ is defined by partitioning the set of subprofiles by aligned epochs, analogous to the procedure for unary profiles given by Equation (13):

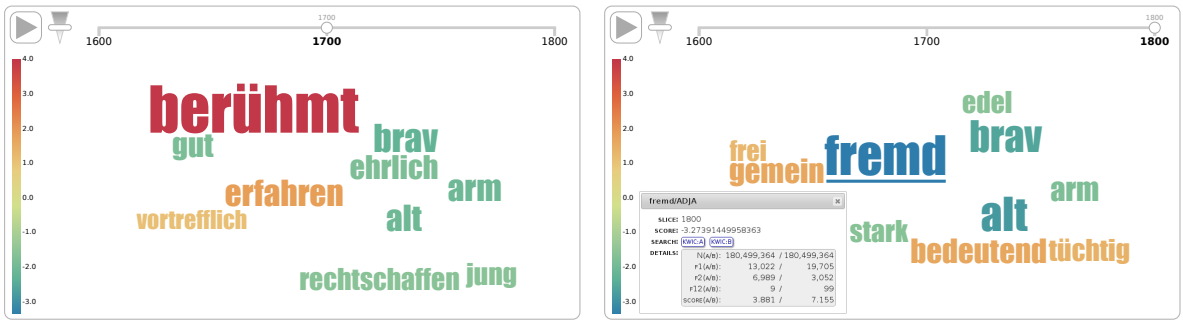
$$\hat{P}_{Q_a \ominus Q_b} \;:\; \mathcal{E}_{a \bowtie b} \to \mathbb{R}^{\mathcal{W}[G]} \;:\; e_{ab} \mapsto \hat{p}_{Q_a \ominus Q_b, e_{ab}} \tag{17}$$

3.6 Output & Visualization

In addition to traditional static tabular display formats suitable for further automated processing (JSON) or import into an external spreadsheet program (TAB-separated text, HTML), the DiaCollo web-service plugin also offers several interactive online visualizations of diachronic profile data for exploratory use. Supported visualization formats include two-dimensional time series plots using the Highcharts JavaScript library, flash-based motion charts using the Google Motion Chart library, and interactive tag-cloud and bubble-chart visualizations using the D3.js library. The HTML and D3-based display formats provide an intuitive color-coded representation of the association score associated with each collocate item, as well as hyperlinks to underlying corpus hits for each data point displayed whenever a DDC search engine for the underlying corpus is available.

[27] The actual comparison score value for $\ominus_{\texttt{havg}}$ is computed as shown in Table 1 as the mean of the harmonic and arithmetic averages of the operand scores in order to avoid singularities due to disjoint operand domains and the associated implicit zero score values.

Figure 2: DiaCollo interactive tag-cloud visualization of the 10 most dissimilar adjectives immediately preceding the noun *Mann* ('man') in the genres "science" (warm colors) and "belles lettres" (cool colors) over the *Deutsches Textarchiv* corpus for the epochs 1700–1799 (left) and 1800–1899 (right).



4 Examples

4.1 Adjectival Attribution: What makes a "man"?

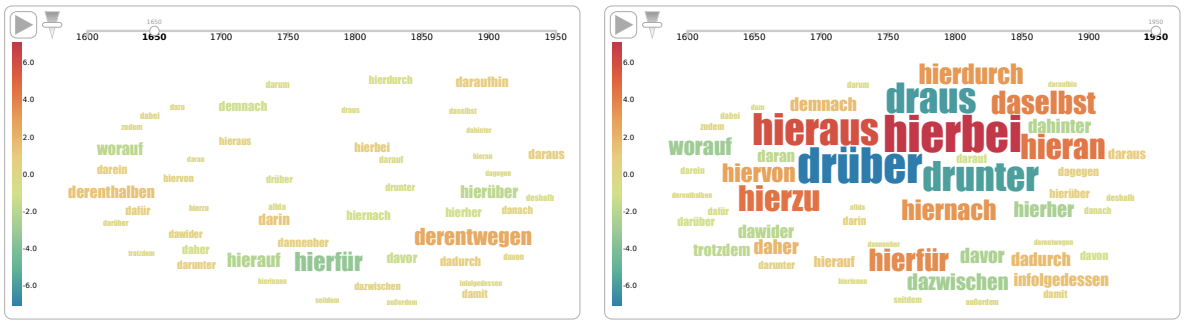
Figure 2 contains example tag-cloud visualizations for a simple cross-genre comparison over the *Deutsches Textarchiv* corpus. The DDC back-end was used to acquire raw frequency counts over 100-year epochs for all attributive adjective lemmata immediately preceding an instance of the noun *Mann* ('man') in the genres *Wissenschaft* ('science') and *Belletristik* ('belles lettres'), respectively. The results were collected as a comparison profile using the absolute difference operation $\ominus_{\texttt{adiff}}$ over log-Dice operand profiles to select the most dissimilar association preferences of *Mann* in the respective genres, as described in Section 3.5. In the tag-cloud visualization mode, absolute magnitudes of score differences are mapped to tag font-size, and the raw score differences are mapped to an intuitive color-scale, with warm tones indicating a relative preference for the "science" genre and cool tones indicating a preference for "belles lettres". As Figure 2 shows, men in scientific texts are more likely to be described as *berühmt* ('famous'), *erfahren* ('experienced'), *bedeutend* ('significant'), or *tüchtig* ('capable'); while men in belles lettres are more likely to be designated *brav* ('well-behaved'), *rechtschaffen* ('righteous'), *arm* ('poor'), or *alt* ('old') – presumably reflecting the properties considered most salient in the context of the respective genres.

4.2 Pronominal Adverbs and Deictic Locality

As closed-class items, pronominal adverbs are good candidates for collocation profiling even in small corpora, since they tend to be highly frequent. Figure 3 shows tag-cloud visualizations for a comparison of pronominal adverb use by text genre over the aggregated *DTA+DWDS* corpus, consisting of the *Deutsches Textarchiv* corpus covering roughly 1600–1900 and the *DWDS Kernkorpus* of 20th century German. Here again, the DDC back-end was used to acquire raw frequency counts for all pronominal adverbs in the genres "science" and "belles lettres", and the absolute difference operation over log-Dice operand profiles was used to select the most prominent dissimilarities.

Most immediately striking is a strong preference on the part of the adverbs *drüber* ('there-over', 'over which', 'about which') and *drunter* ('there-under', 'under which', 'among which') for the literary genre, especially in younger epochs. Closer inspection of the associated corpus

Figure 3: DiaCollo interactive tag-cloud visualization of the 50 most dissimilar pronominal adverbs in the genres "science" (warm colors) and "belles lettres" (cool colors) over the aggregated *DTA+DWDS* corpus for the epochs 1650–1699 (left) and 1950–1999 (right).



hits via the hyperlinks supplied by DiaCollo shows that this preference is due almost exclusively to the colloquial fixed expression *drunter und drüber* ('at sixes and sevens', 'chaotically disorganized'), rather than any alternative independent lexical senses of these items, for which both academic and literary texts tended in later epochs to employ the uncontracted variants *darunter* rsp. *darüber*. Although a number of laboratory situations might accurately be described as "chaotically disorganized", such a state is very much at odds with the scientific ideal of sober, systematic investigation. Taken together with the colloquial flavor of the fixed expression, it is unsurprising that it has seen only minimal use in scientific prose.

The tag-cloud animation further reveals divergent trends in the behavior of local vs. non-local deictic adverbials in the respective genres, visible beginning in the second half of the 18th century. Scientific texts show a strong preference for local deictic adverbials such as *hierfür* ('here-for', 'for which'), *hierbei* ('here-by', 'by which'), and *hieraus* ('here-out', 'out of which') which literary texts lack. The corresponding non-local variants such as *dafür* ('there-for'), *dabei* ('there-by'), and *daraus* ('there-out') show only minimal differences across the target genres, tending to zero in later epochs.

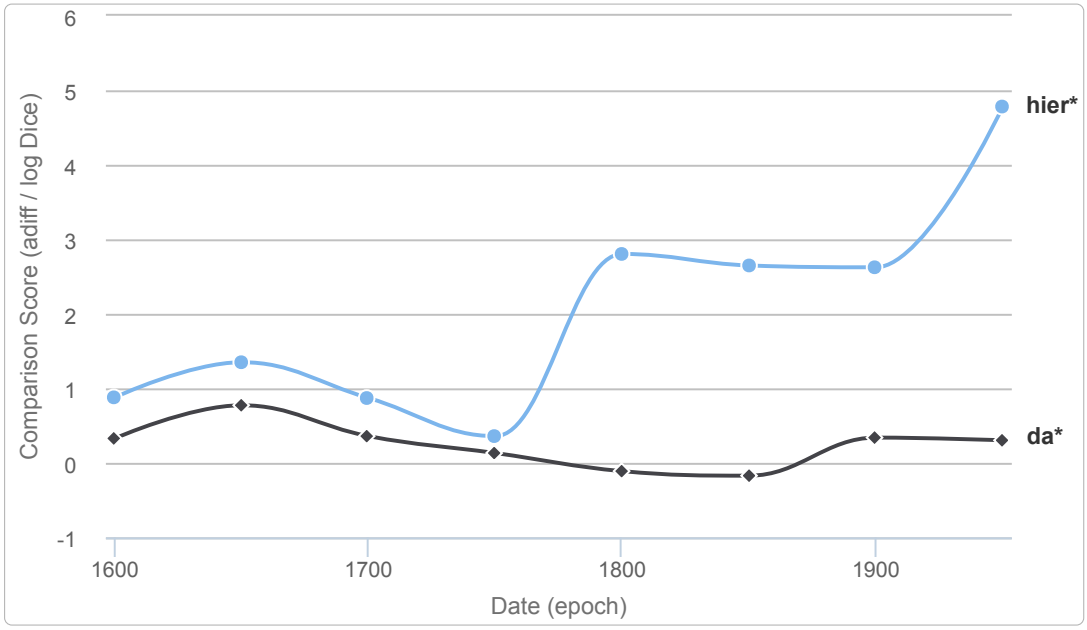
Focusing our attention on the locality of these three pairs leads to the DiaCollo time series plot in Figure 4, in which the inter-genre divergence between the epochs 1750–1799 and 1800–1849 is immediately apparent, which after a plateau of ca. 150 years increases again in the final epoch (1950–1999). An adequate analysis of this phenomenon is beyond the scope of this contribution, but cursory examination of the associated corpus hits indicates that the *hier-* adverbials are used primarily in their epistemic senses,[28] while the corresponding *da-*

[28] I use the term "epistemic" here in its most literal sense 'of or pertaining to knowledge', epistemic readings of (anaphoric) pro-adverbs being those which express a (necessary) logical or epistemic relation between the antecedent proposition and that modified by the pro-adverb (typically a logical consequent), as in (emphasis added):

> *Ursprünglich besagt Wahrheit soviel wie Erschließendsein als Verhaltung des Daseins. Die* ***hieraus*** *abgeleitete Bedeutung meint die Entdecktheit des Seienden.*
> 'Originally, truth implies as much as opening-up as comportment of the Dasein. The meaning derived **from which** is the discoveredness of the entity.' (Heidegger 1927)

If any epistemic modality in the usual linguistic sense is associated with such uses, it would seem to be the necessity of shared knowledge (epistemic state) common to author and reader. Since anaphor resolution in general is often taken to be subject to such constraints (Stalnaker 1974, 2002), no additional lexical characterization of *hier-* adverbials themselves as carriers of epistemic modality need be implied. Such uses may nonetheless be said to be lexicalized (and therefore "senses") to the extent that they impose additional semantic constraints on their antecedents and/or arguments. If on the other hand such ideational usage is taken

Figure 4: DiaCollo time series plot of selected local (*hier-*) and non-local (*da-*) deictic pronominal adverbs (*-für*, *-bei*, and *-aus*) in the aggregated *DTA+DWDS* corpus (1600–1999). Plotted *y* axis values are differences in log-Dice association scores for scientific vs. literary texts; higher values indicate a stronger preference for scientific texts.



variants tend to favor (spatial and/or temporal) locative and telic readings.

If this tendency is representative, the stronger synchronic association of *hier-* adverbials with scientific prose might be explained in terms of the communicative aims underlying the two genres – scientific argumentation relying more heavily on explicit epistemic relations, while narrative prose is more concerned with spatio-temporal and telic exposition. Scientific texts do indeed seem to make more frequent use of epistemic relations, as supported by the genre's stronger association with non-deictic consequentials such as *demnach* ('according to which') and *infolgedessen* ('following from which'). Literary texts on the other hand are more strongly associated with non-deictic spatial and temporal locatives such as *worauf* ('whereupon') and *seitdem* ('since which'), as well as adversative/concessive connectors including *dawider* ('contrary to which') and *trotzdem* ('despite which'). Both Biber et al. (1999) and Herrmann (2013) found that local ("proximate") demonstratives such as *this* and *these* occur more frequently in English-language academic texts than their non-local ("distant") counterparts *that* and *those*: the former study argues that the local variants allow more precise and efficient anaphor resolution by limiting the number of available antecedents, and the latter identifies a strong preference for metaphorical (ideational) uses of local demonstratives in academic text.

I speculate that the shift towards epistemic readings of *hier-* adverbials may have arisen in conjunction with (or perhaps even in response to) their heavier use in academic prose. Use of local propositional deixis in argumentative prose may itself have been further motivated by an active rhetorical strategy on the part of the authors: by using *local* deictic adverbials, the

to be strictly metaphorical (Herrmann 2013), no lexicalization is required and what I have called "epistemic senses" above are simply "epistemic readings."

propositional referents – presumably the arguments' premises – were positioned conceptually "closer" to the reader, implicitly encouraging him or her to accept their validity and thus that of the argument as a whole. On the other hand, the use of non-local deixis in literary contexts may in fact serve to support the reader's deictic shift toward the (fictional) narrative index by drawing his or her attention away from the (real but suspended) deictic center (Galbraith 1995).

5 Conclusion

The formal model for diachronic collocation profiling with query-dependent epoch granularity and attribute collation as implemented in the open-source software tool DiaCollo was introduced and some advantages with respect to conventional collocation discovery software were discussed. In its top-level incarnation as a modular web service plugin, DiaCollo provides a simple and intuitive interface for assisting linguists, lexicographers, and humanities researchers to acquire a clearer picture of variation in a word's usage over time and/or corpus subset. The software's capabilities for detecting genre-sensitive phenomena were demonstrated in terms of two example case studies contrasting the behavior of selected items in the genres "science" and "belles lettres" in diachronic corpora of German. Future work will focus on implementation of additional association score functions and a runtime script interpreter, as well as development of a cross-product profile model and associated visualizations suitable for local collocation network analysis. Publicly accessible DiaCollo web-service instances exist for a number of corpora[29] hosted by the DWDS project at the Berlin-Brandenburg Academy of Sciences, and the DiaCollo source code itself is available via CPAN.[30]

References

P. Baker, C. Gabrielatos, M. Khosravinik, M. Krzyżanowski, T. McEnery, and R. Wodak. A useful methodological synergy? Combining critical discourse analysis and corpus linguistics to examine discourses of refugees and asylum seekers in the UK press. *Discourse & Society*, 19(3):273–306, 2008.

M. W. Berry, S. T. Dumais, and G. W. O'Brien. Using linear algebra for intelligent information retrieval. *SIAM Review*, 37(4):573–595, December 1995. URL `http://www.jstor.org/stable/2132906`.

D. Biber, S. Johansson, G. Leech, S. Conrad, and E. Finegan. *Longman Grammar of Spoken and Written English*. Longman, London, 1999.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of machine Learning research*, 3:993–1022, 2003. URL `http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf`.

V. Brezina, T. McEnery, and S. Wattam. Collocations in context: A new perspective on collocation networks. *International Journal of Corpus Linguistics*, 20(2):139–173, 2015. doi: 10.1075/ijcl.20.2.01bre.

[29] `http://kaskade.dwds.de/~jurish/diacollo2017/corpora`
[30] `http://metacpan.org/release/DiaColloDB`

K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29, 1990.

M. Davies. Expanding horizons in historical linguistics with the 400-million word Corpus of Historical American English. *Corpora*, 7(2):121–157, 2012. URL http://davies-linguistics.byu.edu/ling450/davies_corpora_2011.pdf.

J. Didakowski and A. Geyken. From DWDS corpora to a German word profile – methodological problems and solutions. In A. Abel and L. Lemnitzer, editors, *Network Strategies, Access Structures and Automatic Extraction of Lexicographical Information*, (OPAL X/2012). IDS, Mannheim, 2013. URL http://www.dwds.de/static/website/publications/pdf/didakowski_geyken_internetlexikografie_2012_final.pdf.

I. S. Duff, R. G. Grimes, and J. G. Lewis. Sparse matrix test problems. *ACM Transactions on Mathematical Software (TOMS)*, 15(1):1–14, March 1989. doi: 10.1145/62038.62043.

S. Evert. *The Statistics of Word Cooccurrences: Word Pairs and Collocations*. PhD thesis, Institut für maschinelle Sprachverarbeitung, Universität Stuttgart, 2005. URL http://elib.uni-stuttgart.de/opus/volltexte/2005/2371/.

S. Evert. Corpora and collocations. In A. Lüdeling and M. Kytö, editors, *Corpus Linguistics. An International Handbook*, pages 1212–1248. Mouton de Gruyter, Berlin, 2008. URL http://purl.org/stefan.evert/PUB/Evert2007HSK_extended_manuscript.pdf.

R. T. Fielding. *Architectural Styles and the Design of Network-based Software Architectures*. PhD thesis, University of California, Irvine, 2000. URL https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm.

J. R. Firth. *Papers in Linguistics 1934–1951*. Oxford University Press, London, 1957.

C. Gabrielatos, T. McEnery, P. J. Diggle, and P. Baker. The peaks and troughs of corpus-based contextual analysis. *International Journal of Corpus Linguistics*, 17(2):151–175, 2012. doi: 10.1075/ijcl.17.2.01gab. URL http://www.jbe-platform.com/content/journals/10.1075/ijcl.17.2.01gab.

M. Galbraith. Deictic shift theory and the poetics of involvement in narrative. In *Deixis in Narrative: A Cognitive Science Perspective*, pages 19–59. Lawrence Erlbaum, Hillsdale, NJ, 1995.

A. Geyken. Wege zu einem historischen Referenzkorpus des Deutschen: das Projekt Deutsches Textarchiv. In I. Hafemann, editor, *Perspektiven einer corpusbasierten historischen Linguistik und Philologie*, volume 4 of *Thesaurus Linguae Aegyptiae*, pages 221–234, Berlin, Germany, 2013. URL http://nbn-resolving.de/urn:nbn:de:kobv:b4-opus-24424.

A. Geyken, A. Barbaresi, J. Didakowski, B. Jurish, F. Wiegand, and L. Lemnitzer. Die Korpusplattform des "Digitalen Wörterbuchs der deutschen Sprache" (dwds). *Zeitschrift für germanistische Linguistik*, 45(2):327–344, 2017. doi: 10.1515/zgl-2017-0017.

K. Glazebrook and F. Economou. PDL: the Perl data language. *Dr. Dobb's Journal*, September 1997. URL http://www.drdobbs.com/pdl-the-perl-data-language/184410442.

S. T. Gries and M. Hilpert. The identification of stages in diachronic data: variability-based neighbor clustering. *Corpora*, 3(1):59–81, 2008. URL http://members.unine.ch/martin.hilpert/VNCC.pdf.

K. Gulordava and M. Baroni. A distributional similarity approach to the detection of semantic change in the Google Books Ngram corpus. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 67–71, Edinburgh, UK, July 2011. ACL. URL http://www.aclweb.org/anthology/W11-2508.

H. S. Heaps. *Information Retrieval: Computational and Theoretical Aspects*. Academic Press, Orlando, FL, 1978.

M. Heidegger. Sein und Zeit. In E. Husserl, editor, *Jahrbuch für Philosophie und phänomenologische Forschung*. Neomarius, Tübingen, 1927.

J. B. Herrmann. *Metaphor in academic discourse*. LOT Dissertation Series. Netherlands Graduate School of Linguistics, Utrecht, 2013.

B. Jurish. DiaCollo: On the trail of diachronic collocations. In K. De Smedt, editor, *CLARIN Annual Conference 2015 (Wrocław, Poland, October 14–16 2015)*, pages 28–31, 2015. URL http://www.clarin.eu/sites/default/files/book%20of%20abstracts%202015.pdf.

B. Jurish, C. Thomas, and F. Wiegand. Querying the deutsches Textarchiv. In U. Kruschwitz, F. Hopfgartner, and C. Gurrin, editors, *Proceedings of the Workshop "Beyond Single-Shot Text Queries: Bridging the Gap(s) between Research Communities" (MindTheGap 2014)*, pages 25–30, Berlin, Germany, March 2014. URL http://ceur-ws.org/Vol-1131/mindthegap14_7.pdf.

B. Jurish, A. Geyken, and T. Werneke. DiaCollo: diachronen Kollokationen auf der Spur. In *Proceedings DHd 2016: Modellierung – Vernetzung – Visualisierung*, pages 172–175, March 2016. URL http://dhd2016.de/boa.pdf#page=172.

A. Kilgarriff and D. Tugwell. Sketching words. In M.-H. Corréard, editor, *Lexicography and Natural Language Processing: A Festschrift in Honour of B. T. S. Atkins*, EURALEX, pages 125–137, 2002. URL http://www.kilgarriff.co.uk/Publications/2002-KilgTugwell-AtkinsFest.pdf.

A. Kilgarriff, A. Herman, J. Busta, P. Rychlý, and M. Jakubíček. DIACRAN: a framework for diachronic analysis. In F. Formato and A. Hardie, editors, *Proceedings of Corpus Linguistics 2015*, pages 65–70, UCREL, Lancaster, 2015.

Y. Kim, Y.-I. Chiu, K. Hanaki, D. Hegde, and S. Petrov. Temporal analysis of language through neural language models. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 61–65. ACL, June 2014. URL http://www.aclweb.org/anthology/W14-2517.

C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, 1999.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. URL https://arxiv.org/abs/1301.3781.

F. Moretti. *Distant reading.* Verso Books, 2013.

P. Rychlý. A lexicographer-friendly association score. In *Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN*, pages 6–9, 2008. URL http://www.fi.muni.cz/usr/sojka/download/raslan2008/13.pdf.

E. Sagi, S. Kaufmann, and B. Clark. Semantic density analysis: Comparing word meaning across time and phonetic space. In *Proceedings of the EACL 2009 Workshop on Geometrical Models of Natural Language Semantics*. ACL, March 2009. URL http://www.aclweb.org/anthology/W09-0214.

J. Scharloth, D. Eugster, and N. Bubenhofer. Das Wuchern der Rhizome. linguistische Diskursanalyse und Data-driven Turn. In D. Busse and W. Teubert, editors, *Linguistische Diskursanalyse. Neue Perspektiven*, pages 345–380. VS Verlag, Wiesbaden, 2013. URL http://www.scharloth.com/files/Rhizom_Zeit.pdf.

A. Schiller, S. Teufel, and C. Thielen. Guidelines fur das Tagging deutscher Textcorpora mit STTS. Technical report, University of Stuttgart, Institut für maschinelle Sprachverarbeitung and University of Tübingen, Seminar für Sprachwissenschaft, 1995.

A. Sokirko. A technical overview of DWDS/Dialing Concordance. Talk delivered at the meeting *Computational linguistics and intellectual technologies*, Protvino, Russia, 2003. URL http://www.aot.ru/docs/OverviewOfConcordance.htm.

R. Stalnaker. Pragmatic presuppositions. In M. Munitz and P. Unger, editors, *Semantics and Philosophy*, pages 197–213. New York University Press, New York, 1974.

R. Stalnaker. Common ground. *Linguistics and Philosophy*, 25(5):701–721, 2002. doi: 10.1023/A:1020867916902.

X. Wang and A. McCallum. Topics over time: A non-Markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 424–433, New York, 2006. ACM. doi: 10.1145/1150402.1150450.