Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 6370–6376
Marseille, 20-25 June 2022

The Lexometer: A Shiny Application for Exploratory Analysis and Visualization of Corpus Data

Oufan Hai, Matthew Sundberg, Katherine Trice, Rebecca Friedman, Scott Grimm
University of Rochester
P.O. Box 270096 Rochester, NY 14627, USA
scott.grimm@rochester.edu
{ohai, msundbe2, ktrice, rfried11}@u.rochester.edu

Abstract

Often performing even simple data science tasks with corpus data requires significant expertise in data science and programming languages like R and Python. With the aim of making quantitative research more accessible for researchers in the language sciences, we present the Lexometer, a Shiny application that integrates numerous data analysis and visualization functions into an easy-to-use graphical user interface. Some functions of the Lexometer are: filtering large databases to generate subsets of the data and variables of interest, providing a range of graphing techniques for both single and multiple variable analysis, and providing the data in a table format which can further be filtered as well as provide methods for cleaning the data. The Lexometer aims to be useful to language researchers with differing levels of programming expertise and to aid in broadening the inclusion of corpus-based empirical evidence in the language sciences.

Keywords: R (programming language), Shiny, Data Analysis, Data Visualization, Lexical Analysis, COCA

1. Introduction

Quantitative methods and corpus research have increasingly been used to underpin research in theoretical syntax as well as lexical and formal semantics. At the heart of many theoretical proposals about nominal, verbal or adjectival structure are particular distributional properties, which are argued to correlate with different syntactic or semantic properties. Well-known examples in semantics include the count/mass distinction in nominal structure and the availability of different determiners (Jespersen, 1913), adjectival scalar structure and the corresponding distribution of different adverbial modifiers (Kennedy and McNally, 2005), and verbal aspectual structure and the corresponding temporal/adverbial modifiers (Vendler, 1967). Parallel issues arise for investigating syntactic phenomena and proposals for their syntactic structure, such as the different types of nominalizations and the range of distributional characteristics assumed to correlate with them (Grimshaw, 1990), or the now classic corpus study of the dative alternation in (Bresnan et al., 2007).

The use of corpus data has determined important properties of classes of nouns, verbs and adjectives and helped (in)validate a range of claims, making it of great importance to the field. At the same time, there are few tools that do not require significant expertise in a programming language such as Python or R, as well as additional familiarity with a range of code libraries and packages, such as ggplot and Plotly to aid data visualization. This creates a significant obstacle for making more sophisticated corpus methodologies available to those without substantial programming background, such as undergraduate and even graduate RAs, while at the same time prohibits researchers who are engaged in theoretical syntax or formal and/or lexical semantic research from engaging in any but quite basic corpus work, since the time investment is too great.

We present the Lexometer, an application written in Shiny, a package in R used to build web applications, available at `https://quantitativesemanticslab.github.io`. The Lexometer is capable of cleaning and filtering data and supports a range of common analyses and visualizations of corpus data. The application provides an intuitive graphic user interface (GUI) which allows to greatly accelerate corpus analysis for supported tasks. The goal of the app is to reduce the the amount of time researchers need to spend to write complex programs to perform basic data science tasks in data exploration and visualization. The Lexometer is able to accept all corpus databases in .csv (comma separated values) format and thus can be quite generally applied. At present, our work has focused on using the Contemporary Corpus of American English (COCA) (Davies, 2009) subsequent to processing in an NLP pipeline, which this paper describes. Finally, results produced using the Lexometer can be easily replicated and verified if steps detailing how the results are generated are known. This paper introduces the various functions of the Lexometer and demonstrates some of its possible uses.

2. Related Work

A variety of tools abound for processing and annotating corpora, and many can do this in a very general fashion, e.g., the ANNIS tool (Krause and Zeldes, 2016) and associated pipeline (Druskat et al., 2016) to pick one example. Yet, to our knowledge, most of these tools do not make directly accessible a quantitative analytic component.

Probably the most used accessible interfaces for corpora are web-based interfaces for corpora, such as COCA's interface (https://www.english-corpora.org/coca/). These provide easy access to a wealth of data, but they are also limited in what can be explored and further annotated. In order to gain a more elaborate understanding of a lexical item's distribution, it is usually necessary to go beyond basic collocational frequency or concordance-style analyses. For instance, to understand a lexical item's syntactic distribution, the corpus must be annotated with a layer of syntactic analysis. This is partially achievable in COCA's web-based platform as in included part-of-speech tags, but in practice to take account of, say, all the determiners and adjectives that occur with a particular noun is unwieldy.

More sophisticated corpus studies require (multiple forms of) annotation of corpus data, e.g., parsing, which allow researchers to investigate features of interest for the study at hand. In the typical case, a researcher will extract from a corpus the instances of the particular lexical items at issue for an analysis, and put those through the relevant processing steps, developing a dataframe of the corpus instances and the relevant features relevant for the analysis, e.g., a noun's syntactic position relative to a verb. This is for instance the methodology followed in the study of the dative alternation in (Bresnan et al., 2007). Yet, as discussed above, to develop such corpora, and further to analyze the data within, requires a significant amount of programming expertise, limiting the participation in this sort of research. The Lexometer relies on a previously constructed dataframe, i.e., a 2-dimensional table, but once that is supplied, a user can complete a range of data analysis tasks with little knowledge of the underlying programming language.

3. Annotated Databases for Lexical Investigation

We demonstrate in this section how the Lexometer can be used in the context of a project on nominal semantics which spurred its initial development. While the application was designed for the general case, and nothing hinges on using the particular corpora annotated by our group, it serves as a detailed demonstration of the level of detail in analysis and visualization that can be achieved through the Lexometer. We have designed the application to handle two primary types of dataframes: (i) those containing corpus occurrences of a lexical item along with annotations of properties of those occurrences and (ii) those containing aggregate statistics of a lexical item's distributional (or other) properties.

A database of grammatical behavior of nouns was constructed to support investigating the different grammatical and/or semantic behaviors of nouns. The data derives from the COCA corpus. COCA is a useful resource since it presents a collection of well-balanced texts which are controlled for quality, and does not inject the sort of uncertainty into studies that, say, raw internet data or Twitter data might. This study uses 4 of the 5 genre types in the corpus: *Fiction*, *Popular Magazines*, *Newspaper*, and *Academic*. (We set aside the *Spoken* genre as it results in too many parsing errors.) In total, the database contains over a roughly 350 million word portion of the 450 million word corpus.

This effort included developing an NLP pipeline to process the data and populate a database containing all relevant information. (Further aspects of the methodology described below, including links to code, are discussed in (Grimm and Wahlang, 2021).) First, it was parsed with the CoreNLP suite (Manning et al., 2014), which includes dependency parsing (De Marneffe et al., 2006) that proves critical for efficiently identifying grammatical patterns. Subsequent processing with a Python script extracts from the parsed output all relevant grammatical relations and represents them as features in the database. More concretely, if the output from the dependency parser contains the dependency DET(DOG, THE), then the script will extract the determiner *the* and, then, in the relevant row of the database representing this occurrence of *dog* in the corpus, mark that the determiner was *the*. All potentially relevant information was extracted from the dependency parse, such as position in the clause, modifiers and all other aspects of the grammatical distribution.

Various post-processing steps were taken to insure the quality of data. For instance, an enormous number of words get tagged as a "noun" by the part-of-speech tagger which may have been abbreviations, brand names, or even unusual punctuation. We filtered the nouns that populated the database so as to consist of only the nouns which occur in the CELEX database (Baayen et al., 1996), which is a large and representative sample of standard English vocabulary. (One drawback of this technique is it will exclude more recent innovations like *bling*.) Of the sentences which contained a noun recognized by this criteria, further exclusion criteria were applied, the most important being the exclusion of instances of the noun where it serves as a modifier in a compound, e.g. compounds such as *school bus* were excluded from the analysis of *school*. Further sentences in the corpus were not included in the final database due to limitations of the NLP tools, such as sentences which were too long for the parser or contain html code which make the parser fail. Ideally, an intermediate step would clean the corpus following the work of (Alatrash et al., 2020), but due to time constraints we have left this for future research.

It is worth noting that such a method, while applied to nouns and to the COCA corpus, is very general and could be applied to investigate any part of speech on any corpus. Further, since we employ "Universal Dependencies" (De Marneffe et al., 2014), that is, dependency annotations that are designed to be cross-linguistically comparable, this general strategy can be

applied to a large number of languages in a comparable way.

From this process, we developed two related, but distinct, databases. First, for each noun, a separate dataframe is produced for all the occurrences of that noun. Thus, there is a file *cat.csv* consisting of a dataframe containing all the instances of *cat* in the processed portion of COCA (the rows) along with all the annotations for syntactic distributional features derived from the dependency parse information. We refer to these databases as INDIVIDUAL databases, as they apply to individual nouns. In addition, a GLOBAL database is produced, which aggregates the statistics from all the individual noun files. Thus, in the global database, the row for *cat* will contain summary statistics about all the distributional information summed across the uses in the individual file for *cat*, e.g., a column in the global database is *Definite Article* and for *cat*, one finds the value 58.4%.

These two databases serve distinct purposes. The individual databases permit in-depth investigations of select nouns and also permit further filtering of the data, as discussed in section 4.3.2. Yet, the individual noun databases are less than optimal for comparative purposes, since in practice, as each noun file contains many thousands of rows, when comparing across more than a few individual noun databases ($<$10), the processing and graphic rendering in R becomes quite slow. However, the aggregated statistics in the global database allow to rapidly compare across a much larger set of nouns. In practice, we have found a highly efficient practice is to use the global database to provide a first-pass analysis of the lexical items under investigation, and then follow up with more detailed analysis for each item using the individual databases, where it is possible to clean those databases using exclusion criteria defined in the filtering steps (section 4.3.2).

4. Lexometer: a Shiny Application

The Lexometer is a interactive web application developed using Shiny and R. The central goal is to gather together functionalities for common tasks for data cleaning and filtering, exploratory data analysis, and data visualization and to make these functionalities available through interactive features such as check boxes and drop-down lists in order to circumvent the need for developing expertise in, e.g., R syntax, thereby accelerating the inclusion of young researchers, and those with less background in quantitative research, into data-based approaches to linguistic analysis. The Lexometer is able to perform tasks as simple as grouping nouns based on categories and as complex as graphing multiple variables on the same graphs. We have purposely limited its functionality as far as the inclusion of more sophisticated statistical modeling techniques, even basic ones such as regression: These techniques require expertise in, e.g., model assumptions, and we assume researchers who apply such models will be conversant with a programming language that implements them. Even in these cases, we have found in practice the use of the Lexometer greatly accelerates the preparation of data for studies involving statistical techniques requiring further programming in R or Python.

As a simple demonstration, we walk through an use of the Lexometer to explore nouns in terms of the count/mass distinction, the propensity for certain nouns, notably those referring to substances, to disallow use of the plural markers and many quantifiers (**sand-s*, **five sands*). This stands in contrast to other nouns, notably those referring to concrete and well-defined objects, which accept plural marking and the full range of quantifiers that rely on counting (*dog-s*, *many dogs*). We proceed by subsetting and filtering the data to the set of nouns and features of interest, perform some basic visualization, and show how we can use the capacities to do fine-grained cleaning and correction on individual noun databases.

4.1. Data filtering and subset generation

The databases elaborated in section 3 contain many thousands of nouns and are annotated for over 200 features, thus to address any particular research question, the relevant nouns must be selected and the data must be filtered for the relevant information for the different features to be brought into the analysis. These processes, data filtering and subset generation, are performed in the "Select Nouns and Filter Data" tab for both "Global Noun Database" and "Individual Noun Database" tabsets.

Researchers may specify one or more subsets of nouns. For our running example of a researcher examining the count/mass distinction, a researcher may create one subset for "substance" nouns, specifying *mud*, *oil*, *sand*, and *water*, and one subset for "object" nouns, specifying *car*, *cat*, *chair*, and *dog*. Later in the analysis, these subsets can be compared in plots. When an adjustment to the nouns in the subsets needs to be made, such as adding a noun, the current plot will be automatically re-rendered to adjust for the change.

To help researchers easily produce the aggregates of data from their databases, the Lexometer offers options for users to build subsets based on user-defined constraints on the database. The constraints can be listed values, values included or excluded by a grep pattern (grep is comprised of a small set of UNIX commands that use regular expressions to search input files for a search string), or numerical filtering with greater-than or less-than values. For example, if a researcher is interested in the determiners that co-occur with a set of nouns, after selecting the nouns, the researcher could also constrain the values returned for determiners, e.g., by listing the determiners of interest (*the*, *a*, *an*, etc.) to be returned, or by specifying through grep pattern-matching all determiners with a word-initial 'th' (also returning *the*, but not *a*, etc.). Further, these can be

constrained to be of a certain numerical value, e.g., return nouns with greater than 10 instances of *the*, or less than 50% of occurrences with *the*. In practice, these methods for filtering the data have been sufficient for manipulating the data as needed for exploratory data analysis purposes.

4.1.1. Noun Selection: Preset Subsets

For global noun databases, we have included the ability to pre-define multiple noun subsets to streamline the process of generating noun sets repeatedly examined by the researcher. Some preset subsets defined for our group, such as *Animals* or *Pluralia Tantum*, are shown in Figure 1. Selecting these options would pre-fill some nouns in the selected category in the Select Nouns *textInput* field.

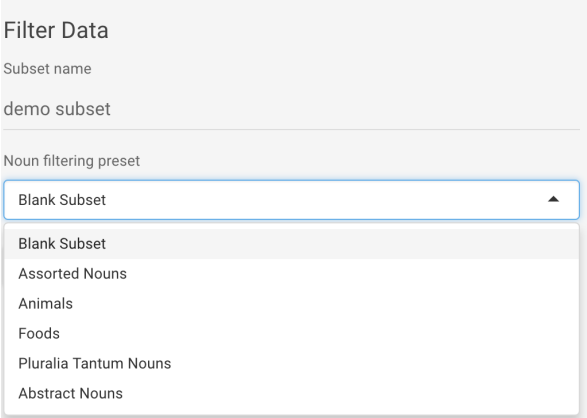

Figure 1: Subset presets for Data Selection

4.1.2. Nouns Selection: Blank Subset

The researcher can also select whichever lexical item(s) that are to be investigated. Here the option "Blank Subset" is chosen, and users have more options to customize the nouns they wish to include along with a number of other variables (see Figure 2). The nouns are separately entered in the "Select Nouns" *textInput* field at the top of the "Select Nouns and Filter Data tab" for the Global Noun Database are passed into the "Noun" variable as shown in Figure 2. Further variables are available to be added once the current variable has been specified.

4.2. Data Visualization Functionalities

We have designed the Lexometer to provide quick access to common data visualization methods, such as bar charts, simple two-variable comparison, group-point graphs and co-occurrence graphs. We have included a wide range of options for customizing the graphics. For instance, to suit the varying lengths of the names of the column data that users may provide, we have included options to adjust the styles of the x-axis labels, e.g., rotating the labels by 90 degrees, 45 degrees, etc. Furthermore, users may adjust the colors of their

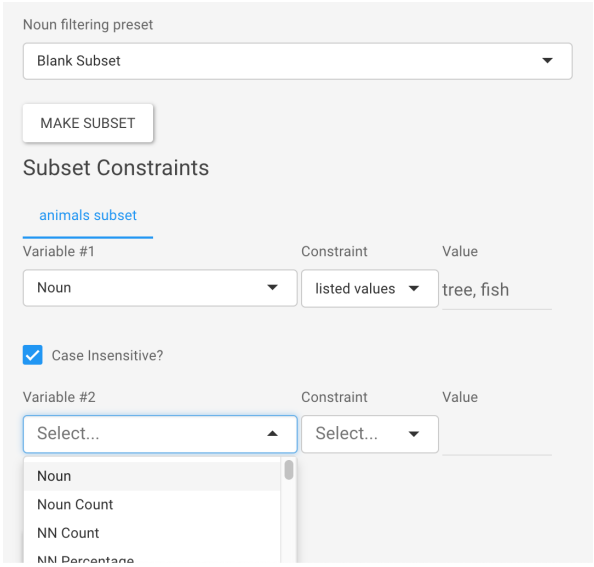

Figure 2: Data selection for Blank Subsets.

graphs to suit their needs and generate the graph that best presents their research results. After finalizing a graph, the user can click a button to download the graph as a .pdf or .png file.

Finally, a practical difficulty is often encountered in that rendering graphics can be quite slow once many nouns/variables are involved. A checkbox allows the user to pause graph rendering which prevents the wasting unnecessary resources caused by constant re-rendering of graphs when users are adjusting their graphing settings. We now review some of the visualization capabilities.

4.2.1. Plotting Bar Charts

Bar chart plotting is one of the functions that we anticipate users to engage with the most to assess distributional tendencies. After users have defined a subset via the Select Nouns and Filter Data Tab discussed in section 4.1, users choose the columns they wish to graph. The columns are read from the Global Noun Database input. Here too we add functionality to preset constellations of variables that the user may wish to repeatedly graph, again streamlining the process. In our work, we have developed three groupings of variables: "verb related", "determiner related", and "plurality related."

Figure 3 shows a bar plot developed in the Lexometer to investigate the count/mass distinction. The user begins with the Global Noun database where two subsets have been defined, one representing 'objects' and one 'substances', as discussed in section 4.1. The user has selected the preset variable grouping "plurality related", and has further removed certain of the variables not pertinent for the analysis by deleting them from the field *Columns to graph* shown in the left panel of Figure 3 and then added the column 'Indefinite Article'. The labels on the x-axis overlapped, therefore the user

selected the option to rotate them 90 degrees so that they are legible. As would be expected, the user is able to verify that these two groups of nouns behave differently as to what types of distributional patterns hold for plurality related variables: 'objects' appear with plural forms and indefinite articles, while 'substances' primarily appear as bare singulars. The researcher can then explore individual differences through the individual noun databases.

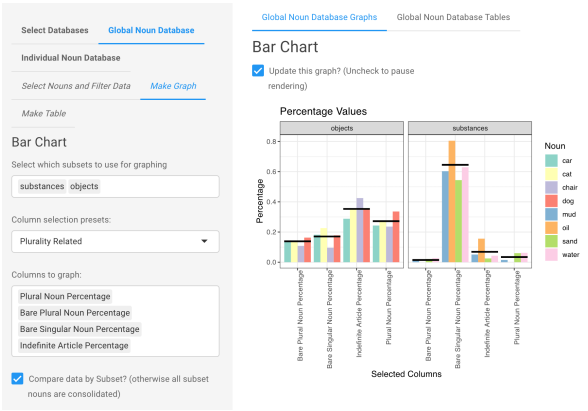

Figure 3: A bar plot contrasting distributional features of count and mass nouns

When graphing with the Individual Noun Databases, users choose a noun or nouns in the "Select Nouns" *textInput* field, but do not need to create subsets for the graphing functions to work. In the "Individual Noun Database" tabset, users are able to plot column breakdown graphs, gives a visual representation of the data in that column. The "columns" available for selection in the drop-down list are data read from the Individual Noun Database. In figure 4, the Individual Noun database *beetle* is chosen and the column "Relation to Verb" is selected. The bar chart presents the percentages of relations like subject, object, and passive. Again, this graphic can be customized in its presenation of labels, color, and so forth.

We now show two other useful capabilities for comparing lexical items against a single variable, group-point comparison and co-occurrence graphs. We then turn to two variable comparison.

4.2.2. Plot Group-point Comparison Graphs

The Group-point Comparison Graph plots the average of the subset on a line. In addition, the data points for each element in the subset are plotted on the same line to enable easier comparison with the group average. Here we demonstrate in Figure 5 with the preset 'Animal' nouns and compare their occurrences with the part-of-speech tag for plural nouns (NNS), measuring the propensity of the different nouns to occur in the plural.

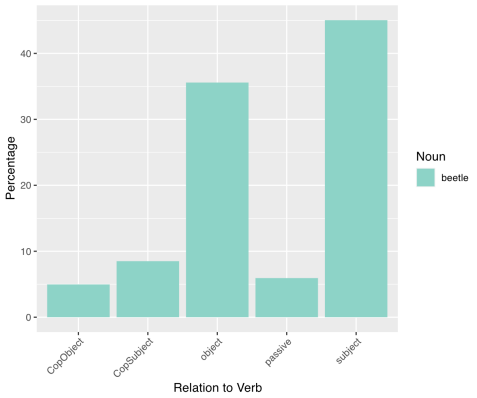

Figure 4: Individual Noun Database column breakdown graph with *Relation to Verb* column selected

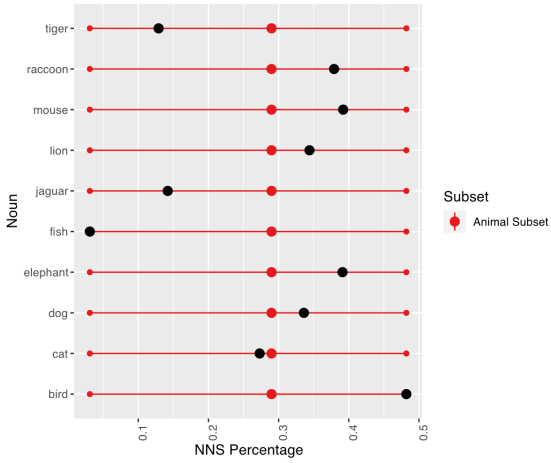

Figure 5: Group-point graph

4.2.3. Plot Co-occurrence Graphs

The Co-occurrence graph can be plotted after the user selects a subset and then select two columns to compare their co-occurrence. As shown in Figure 6, the resulting graph indicates the co-occurrence of nouns and their relation to verbs. The intensity of colors in the graph signals the extent of co-occurrence, here that nouns occur most often in subject position.

4.2.4. Plot Two Variable Comparison Graphs

Two Variable Comparison graphs plot one variable (column) extracted from the Global Noun Database against another variable. The resulting graph is a scatter point plot. Here we demonstrate with the preset 'Animal' nouns and compare their occurrences with the part-of-speech tag for singular nouns (NN) or plural nouns (NNS), which allows to compare if nouns have a propensity to occur in the singular or the plural.

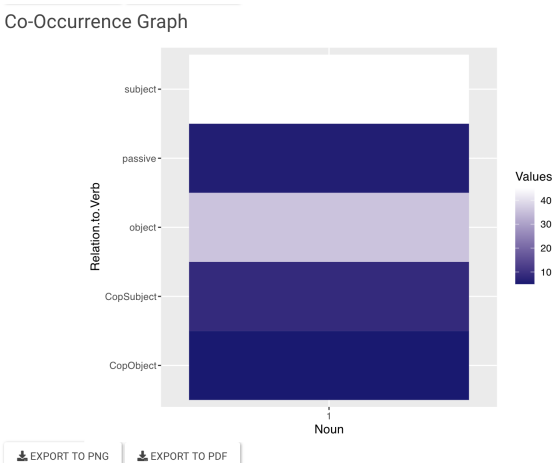

Figure 6: Co-occurrence graph

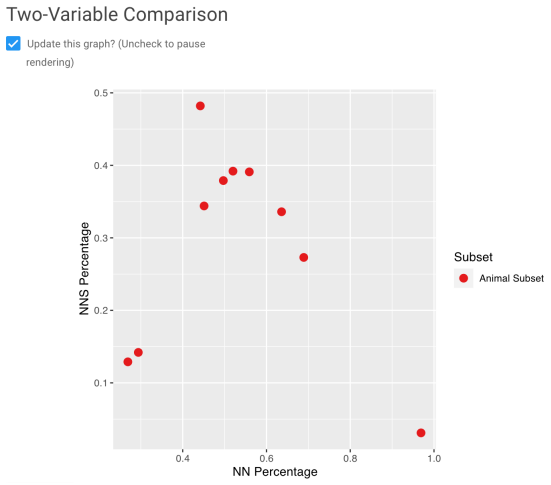

Figure 7: Two variable comparison graph

4.3. Table Generation

The Lexometer is also able to generate tables based on selected columns of the noun subset created by the user. For the Global Noun database, users can create table views of different slices of the aggregate statistics. In the case of the Individual Noun databases, users can examine individual corpus occurrences and their annotated properties. All tables generated within the Lexometer can be downloaded as .csv files by clicking on a "Download CSV" button (in parallel to how PDFs of graphs can be downloaded).

4.3.1. Generating Tables From the Global Noun Database

Users first need to specify the subset they wish to investigate; then, they can select the columns to include in their table. In the example below, the preset "Animal subset" shows ten types of animals and includes five columns in the table — Noun, NNS percentage, NN percentage, Celex Uncountable, and Celex Countable — which track plural and singular uses in COCA against whether these nouns were classified in the CELEX database has have countable or uncountable uses.

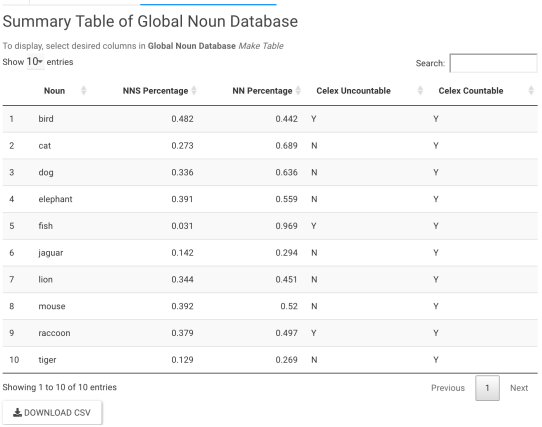

Figure 8: Table of Selection of Global Noun Aggregate Data

4.3.2. Generating Tables of Individual Noun Uses

As mentioned, tables generated from the Individual Noun database provide an opportunity to get a fine-grained view into the data and potentially detect errors and clean the data. Figure 9 shows a table generated for the occurrences of the non-countable noun *traffic* from its Individual Noun database. Here the researcher suspects that many occurrences of *traffic* are parts of compounds which should have been excluded from the analysis. The user selects the columns 'Sentence Fragment' (giving the immediate context of the noun), 'Compound Head' and 'Relevant Dependencies'. The third line of the table shows that indeed, one instance of *traffic* is the compound *traffic jam*, which should be excluded. Use of the table function can be combined with the filtering functions, so the user can return to the filter settings, set the 'Compound Head' variable to exclude any non-empty occurrences, which automatically will be updated in the table. The user can then download the resulting .csv file and then proceed to use the cleaned data for subsequent analyses.

5. Conclusion

This paper has presented the Lexometer, an application for making exploration of quantitative linguistic data more accessible and in the process streamlining an accelerating many common functionalities. Our goal is to be able to involve a greater portion of the language science community in corpus and quantitative work even if they have not yet developed sophisticated programming skills in, e.g., R and Python. The Lexometer simplifies a range of tasks, from cleaning data to making plots, which we hope will spur greater participation in the broader language community to adopt quantitative methods as a part of their research portfolio. In future work, we expect that adadpting the Lexometer to

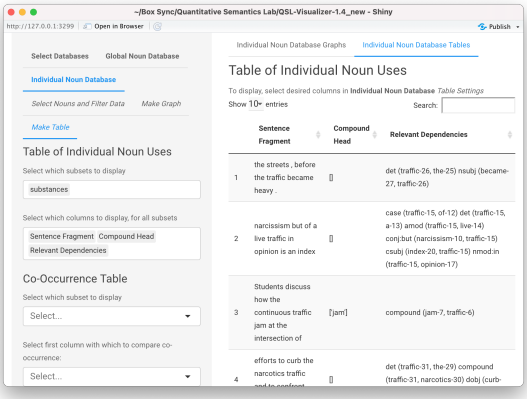

Figure 9: Table of Selection of Individual Noun Aggregate Data

databases beyond those developed in our research will aid us to generalize and improve many of the functionalities as well as the UI.

6. Bibliographical References

Alatrash, R., Schlechtweg, D., Kuhn, J., and im Walde, S. S. (2020). CCOHA: Clean corpus of historical American English. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6958–6966.

Bresnan, J., Cueni, A., Nikitina, T., and Baayen, R. H. (2007). Predicting the dative alternation. In *Cognitive foundations of interpretation*, pages 69–94. KNAW.

De Marneffe, M.-C., MacCartney, B., and Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In *LREC*, volume 6, pages 449–454.

De Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C. D. (2014). Universal Stanford dependencies: A cross-linguistic typology. In *LREC*, volume 14, pages 4585–92.

Druskat, S., Gast, V., Krause, T., and Zipser, F. (2016). corpus-tools.org: An interoperable generic software tool set for multi-layer linguistic corpora. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4492–4499, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Grimm, S. and Wahlang, A. (2021). Determining countability classes. In *Things and Stuff: The Semantics of the Count-Mass Distinction*, pages 357–377. Cambridge University Press.

Grimshaw, J. (1990). *Argument structure.* MIT Press.

Jespersen, O. (1913). *A Modern English Grammar on Historical Principles, Part II: Syntax*, volume 1. Allen & Unwin, London.

Kennedy, C. and McNally, L. (2005). Scale structure, degree modification, and the semantics of gradable predicates. *Language*, pages 345–381.

Krause, T. and Zeldes, A. (2016). ANNIS3: A new architecture for generic corpus query and visualization. *Digital Scholarship in the Humanities*, 31(1):118–139.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Vendler, Z. (1967). *Linguistics in Philosophy*. Cornell University Press, Ithaca, NY.

7. Language Resource References

R.H. Baayen and R. Piepenbrock and L. Gulikers. (1996). *CELEX2*. Linguistic Data Consortium.

Davies, Mark. (2009). *The 385+ million word Corpus of Contemporary American English (1990–2008+): Design, architecture, and linguistic insights*.