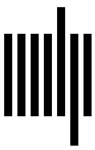
The Open Handbook of Linguistic Data Management

Edited by: Andrea L. Berez-Kroeker, Bradley McDonnell, Eve Koller, Lauren B. Collister

Citation:

The Open Handbook of Linguistic Data Management
Edited by: Andrea L. Berez-Kroeker, Bradley McDonnell, Eve Koller, Lauren B. Collister
DOI: 10.7551/mitpress/12200.001.0001
ISBN (electronic): 9780262366076
Publisher: The MIT Press
Published: 2022



The MIT Press

49 Managing Phonological Data in a Perception Experiment

Rory Turnbull

1 Introduction

In this chapter, I'll be walking through the data management workflow used in the study reported by Turnbull and Peperkamp (2017). This study investigated phonological priming in French. Simplifying somewhat, the experiments involved participants hearing a prime word followed by a target word; the task was to decide whether the target was a lexical word or a non-word (a *lexical decision* task). The experimental manipulation was the degree and type of phonological overlap between the prime and target. For example, the target word *bac* /bak/ "tray" could have as a prime any of *bac* /bak/ "tray," *sac* /sak/ "bag," *banque* /bãk/ "bank," *baffe* /baf/ "slap," or *mangue* /mãg/ "mango." If you're interested in the theoretical background and the implications of our results, I refer you to the original paper. For the rest of this chapter, I'll refer to the Turnbull and Peperkamp (2017) experiments as "TP17."

I've chosen the TP17 experiments for this data management use case as, methodologically speaking, they're rather mundane. They use standard methods for perception experiments in phonology, phonetics, and psycholinguistics, and it should be fairly straightforward to find correspondences between the data components of these experiments and many others.

I assume that you, the reader, have a basic familiarity with experimental methods in linguistics. Nevertheless, I've tried to avoid jargon and be clear and explicit in this chapter. If I've been successful, then this chapter should be easy to follow even if you're unfamiliar with these methods. (And if so, hello, welcome to the world of experimental linguistics!)

1.1 What is a phonological perception experiment?

For the purposes of this chapter, *phonological perception experiment* is intended as a big-tent term for an experiment with auditory linguistic stimuli and non-auditory responses, where the focus of investigation relates to the form of the stimuli, rather than their meaning. Crucially, the research question in these studies relates to the mechanisms of perception and related processes. This includes artificial language learning experiments (e.g., Finley 2011, 2012; Katz & Fricke 2018); speech intelligibility studies (e.g., Warren et al. 1995; Youngdahl et al. 2018); word comprehension eye-tracking tasks (e.g., Brouwer, Mitterer, & Huettig 2012; Ito & Speer 2008); phoneme monitoring tasks (e.g., Damian & Bowers 2010; Hay, Drager, & Gibson 2018); phoneme identification tasks (e.g., McGurk & MacDonald 1976; Mitterer 2006); prosody perception tasks (e.g., Cole, Mo, and Hasegawa-Johnson 2010; Turnbull et al. 2017); and many others. Many of the best practices associated with these kinds of experiments can be extended to similar but different tasks, such as (non-linguistic) psychophysical perception studies (e.g., Ladd et al. 2013), or perception studies with a production component (e.g., Wiener & Turnbull 2016).

Still, even this broad definition excludes perception research on signed languages, which use a visual rather than auditory modality. I suspect that many of the issues discussed in this chapter will map reasonably well to experiments on signed languages, such as Dye and Shih (2006), substituting "auditory" for "visual" as appropriate. Nevertheless, the rest of the discussion in this chapter will refer exclusively to studies of spoken language.[1] For data management of sign language data in various contexts, see Hou, Lepic, and Wilkinson (chapter 40, this volume), Hochgesang (chapter 30, this volume), Crasborn (chapter 39, this volume), and Palfreyman (chapter 21, this volume).

1.2 Chapter overview

In experimental linguistics, the term *data* is often used synonymously with the recorded responses from the

experiment. "Let's look at the data" is usually an invitation to examine the experiment's results. However, the entire research process, from experiment conceptualization through to sharing the results involves a great deal of data and data generation. Each of these kinds of data have different management needs.

In this chapter, I discuss in turn the following topics: organizational principles; stimuli; participant information; experimental script(s); experiment responses; statistical analysis script(s); and storing, sharing, and citing. For each of these, I first describe the management we employed (or, in some cases, ought to have employed!) for TP17, followed by more general considerations for other projects.

A common thread running through these topics is that of documentation. "Always include a readme file" is a common (but often ignored!) edict in software engineering, and the same is true of each step of the experimental research process. Work as though another person is going to have to see and understand everything you've done. This other person might be a collaborator, a supervisor, or—the most likely case—your future self. Make your future life easier and document as much as you can.

Your documentation needs to be *interpretable*; the main points should be easily gleaned from the opening. Even if you have no plan to share any of these details, the documentation you write now will surely help your future self when you return to the project in a month's time, a year's time, or even a decade's time. The whole point of the documentation is to make life for your future self (and other future readers) easier.

Your documentation also needs to be *accessible*. Have you ever tried to open a WordPerfect file from 1995 on a modern computer? While not impossible, it's a lot more difficult than it has any right to be. Have you ever opened a file and all the formatting is messed up? The phonetic symbols and non-roman characters have been transformed into gibberish like *æ–‡å—åŒ–ã'*? Your documentation ought to be as future-proof as possible. For that reason, I recommend using plain text files (.txt), with a Unicode standard encoding such as UTF-8 (eight-bit Unicode Transformation Format).

Writing this kind of documentation is a curious exercise, especially when your target audience is yourself. Surely you understand what this code is doing, how these stimuli are organized, why you chose this particular sampling frequency? My personal experience suggests that I greatly overestimate the intelligence of my future self.[2] There's an adage coined by Douglas Hofstadter (1979) called *Hofstadter's Law*: "It always takes longer than you expect, even when you take into account Hofstadter's Law." I hope I do not do too much damage to this law by suggesting a related law: your documentation always needs more detail than you expect, even when you've taken this law into account.

With that framing in mind, let's begin.

2 Organizational principles

You're beginning a new project. You create a new folder on your computer and then add a few files to that folder. Maybe you have a couple of documents with notes, some saved e-mails with ideas from collaborators, and a couple of interesting papers that are relevant to the project. As the project develops, you add more and more files, until eventually it's a sprawling mess of files, with no organization, no clear hierarchy, and not even a way to know which files are the most up-to-date. Does this sound familiar? If so, you're not alone, and treatment is available.

Let's consider how we can improve on this method. First, clarity through structure. Each distinct form of data should have its own folder within the main project folder. Second, clarity through documentation. Each folder should have a readme file that explains the contents of the directory. The main project folder should have its own readme file for the entire project. Each of the kinds of data discussed in this chapter—stimuli, experimental script, experimental responses, statistical analysis—should have its own folder, and each folder its own readme. Depending on how you organize your files, you may also want a folder of relevant papers, a folder for conference presentations based on this data, a folder for the manuscript you're writing, and others.

Using this method is no guarantee that the project will be perfectly organized. Changes in ideas and plans often necessitates a change in structure. Keeping readme files up to date requires attention. But using this method will make it less likely that your project folder will descend into chaos. With these organizational principles outlined, let's turn to our discussion of different data types.

3 Stimuli

The *stimuli* are an important part of any experiment, and often something that are relatively easily shared with

others. I distinguish three main kinds of stimulus data: the (master) stimulus list, the stimulus media, and the presentation lists. Finally, I end with a cautionary note about character encodings.

3.1 Stimulus list

Also called a "master" list or "grand" list, this is a list of all the stimuli used in the experiment. For TP17, developing this list took many hours of work, as we wanted French words of consonant-vowel-consonant, or CVC, shape with several phonological competitors to serve as primes. Ultimately this work involved carefully combing through the French lexical database Lexique (New et al. 2001) to find appropriate words. Lexique includes frequency counts and other psycholinguistically relevant information, which we incorporated into our list. We also had to create nonwords with plausible real word competitors and obey various constraints of counterbalancing.

Recall the design of this experiment—participants heard a prime word, followed by a target item that may or may not be a word and that has some particular phonological relationship to the prime. In designing the stimuli, we had multiple possible primes for each target. As can be seen in figure 49.1, the first target is *bac*, which could be preceded by any of *bac*, *sac*, *banque*, *baffe*, or *mangue* as primes. Note that some combinations were impossible and that there are empty cells: there are no onset competitors or vowel competitors for *bègue*.

While the organization in figure 49.1 is pleasant to look at and relatively easy to understand, it is not so helpful for generating lists of stimuli for presentation to participants. It also does not show the word metadata, such as frequency, morphological information, number of neighbors, and so on. Indeed, to get this metadata, we'd need at least an additional five columns per kind of metadata.

For this reason, our final stimulus list was a spreadsheet file with one row per stimulus word. An excerpt is shown in figure 49.2. This organization loosely follows the principles of "tidy data" (Wickham 2014).[3] Each set of target and prime words is given a unique identifier (ID) number, enabling the data to be transformed into the structure shown in figure 49.1 without fuss. Because each word has its own row, there is only a need for a single column for each of our kinds of metadata. (The metadata here was lifted straight from Lexique.)

The conditions in the Condition column are prefixed with the letters *A* through *F*; rather than writing "Target," we've written "ATarget"; instead of "Homophone," we have "BHomophone"; and so on. This was solely so that when we sorted the spreadsheet by the Condition column, we'd see the conditions in the order we wanted them. This is helpful to us, the spreadsheet makers, but not helpful to spreadsheet readers without explanation. Indeed, more confusingly, the terms used in this spreadsheet are different from those we ended up using in figure 49.1 and in the final paper (where we used, e.g., "_VC" rather than "OnsetMP"). Consequently, the readme file in the stimulus folder explains the correspondences between the published paper and the spreadsheet.

Other details in the readme file include how to interpret the details from Lexique, such as the phonological transcriptions and the morphological parse.

3.2 Stimulus media

This category includes audio recordings and any other media used, such as video recordings or images. For TP17, the prime words and target words were recorded by separate talkers. We used a custom-made MATLAB[4] script to facilitate our recordings. This script presented the talkers with the stimulus words to be recorded and made a single Waveform Audio File (WAV) file of the microphone input during that particular word. The WAV file was automatically trimmed to remove silence around the edges. The recordings were then amplitude-normalized via a Praat[5] script (Boersma & Weenink 2020).

We didn't report most of this detail in our paper, as the particulars of *how* we segmented the recordings aren't really germane to the methods section and don't impede replicability. But for internal purposes, we documented each stage of this, including what scripts were

Target	CVCPrime	_VCPrime	C_Cprime	CV_Prime	UnrelatedPrime
la bague	une bague	une vague	un bogue	plus basse	qu'il mange
un bac	la bac	le sac	une banque	une baffe	la mangue
un bègue	le bègue			elle baigne	la soeur
plus belle	très belle	le gel	une balle	une benne	les pognes
je me baisse	il baisse	une caisse	un bus	la base	elles fouillent

Figure 49.1
Example sets of prime/target pairs from TP17.

WordTrial	Condition	ID	Phrase	Orth	Phon	Class	Gender	Number	Frequency	Morpholog	Neighbours
Word	ATarget	1	un	bac	bak	NOM	m	s	9.03		32
Word	BHomophone	1	un	bac	bak	NOM	m	s	9.03		32
Word	COnsetMP	1	le	sac	sak	NOM	m	s	105.96		27
Word	DVowelMP	1	une	banque	b@k	NOM	f	s	70.79		10
Word	ECodaMP	1	une	baffe	baf	NOM	f	s	1.41		27
Word	FUnrelated	1	la	mangue	m@g	NOM	f	s	0.73		13
Word	ATarget	2	je	file	fil	VER			36.47	imp:pre:2s	27
Word	BHomophone	2	je	file	fil	VER			36.47	imp:pre:2s	27
Word	COnsetMP	2	de	Gilles	Zil	NOM	m	p	2.43		22
Word	DVowelMP	2	la	foule	ful	NOM	f	s	25.95		19
Word	ECodaMP	2									
Word	FUnrelated	2	pas	russe	Rys	ADJ		s	24.85		21
Nonword	ATarget	152	les	duche	dyS						
Nonword	ECodaMP	152	la	dune	dyn	NOM	f	s	1.55		13
Nonword	ATarget	153	un	veauf	vof						
Nonword	COnsetMP	153	tu	chauffes	Sof	VER			0.42	ind:pre:2s;	10
Nonword	ATarget	154	la	guèffe	gEf						
Nonword	COnsetMP	154	des	chefs	SEf	NOM		p	15.46		11

Figure 49.2
Extract from the grand stimulus list from TP17.

used, and we retained a copy of the raw recordings in a separate subfolder. The relevant scripts themselves were kept in their own subfolder, as they are a key part of the project and should not be separated from it.

TP17 was fairly ordinary terms of stimulus media, but for speech perception research involving artificially generated or manipulated stimuli, there are a lot of steps here to document. Your documentation here should essentially be a how-to guide so that someone with only limited technical expertise can reproduce your steps to arrive at functionally equivalent stimulus media.

3.3 Presentation lists

This category includes the actual lists involved in the experiment, if necessary. That is, these are the lists of stimuli, in order, that are presented to each participant. For within-subjects designs, these lists are essentially reordered (pseudorandomized) versions of the grand stimulus list. For between-subjects designs like those used in TP17, generation of these lists can involve a lot of careful counterbalancing. We ended up with four distinct lists, each a proper subset of the master list.

3.4 A note about character encodings

It would be remiss of me at this point to fail to mention character encodings (see also Han, chapter 6, this volume). Character encodings are the techniques used to represent symbolic data—for example, in Morse code, the letter *A* is encoded as "dot dash," while in ASCII,[6] *A* is represented as the number "65." The Lexique database downloadables are encoded in ISO 8859-1, a character encoding designed for western Europe. If the files are read as if they're UTF-8 (a common modern Unicode standard and the default encoding for many operating systems), several accented characters will display incorrectly. Therefore, at an early stage in constructing the stimulus lists, we converted the Lexique database to UTF-8 to avoid other problems. However, care should be taken to ensure that the experimental presentation software is capable of correctly displaying the encoding you're using. Indeed, some older software does not work well with Unicode. Encoding problems are frustrating, so plan ahead to save a headache, especially if you're working with non-Latin orthographies.

4 Participant information

Experiments have participants, and we need to collect information on those participants. This information comes in two kinds: legally required information and documentation, and research-relevant information and documentation. The first kind satisfies the legal (and ideally ethical) obligations we have to our participants, while the second kind of information is driven by some research-related motivations.

I'm deliberately excluding from this category any *behavioral responses*, that is, the results of the main task of the experiment; instead, this category is usually

information *about* the participants and could perhaps be termed "participant metadata."

4.1 Legal paperwork

The details here will differ depending on your jurisdiction, but this will likely involve, at a minimum, documentation of informed consent. Other aspects may include debriefing forms, compensation receipts, evidence of prior ethical approval from a board (e.g., an institutional review board in the United States), and so on. These documents are, in some sense, the most important pieces of data for your entire project, as there can be stiff administrative and legal consequences if things are out of order. For TP17, we collected paper consent forms with signatures, along with documentation of receipt of compensation.

4.2 Research paperwork

These are data relating to participants that is not legally required, nor is it part of the experiment proper, but still constitutes useful information. I've called this "paperwork" as it's often collected in the form of a paper questionnaire, but many researchers choose to collect this via computer.

For TP17, the research paperwork consisted of a questionnaire about the participant's language background. The responses to these questionnaires were then collated into a digital spreadsheet. Note that individual participants were identified in the research paperwork solely by a unique ID number. No names were used here, thus helping with anonymity requirements. This ID was also used to link participant metadata to participant responses in the statistical analysis.

Other elements of research paperwork include a list of all participants (facilitating the application of exclusion criteria), scheduling information, and all logistical details related to participants actually participating.

5 Experimental script

By *script* I mean the computer file(s) that will present the experimental stimuli and record responses. In TP17, the we used the PsychoPy experimental presentation software (Peirce et al. 2019), and our script is therefore constituted of the PsychoPy code file, the presentation list comma-separated values, or CSV, files it reads, and the folders of WAV files it accesses to present the stimuli.

There are many options in the world of experimental presentation software, each with distinct implications for data management. One advantage of PsychoPy is that it's open source, so anyone can use it and see the details of the script without needing to install expensive software. Had we used a closed-source, proprietary system (such as, e.g., E-Prime[7] or SuperLab[8]), it would be harder for researchers in subsequent decades to examine exactly what the script did, especially if those companies do not exist any longer. While this might sound far-fetched, this scenario can and has happened. SensoMotoric Instruments (SMI), a prominent maker of eye-tracking hardware, was acquired by Apple in 2017. Apparently, the corporate leadership at Apple wanted a new direction, and as of this writing (early 2019), SMI no longer makes new trackers or repairs old ones. For existing users of SMI products, their customer support is still active, but it's not clear whether it always will be. If, ten years from now, you are trying to examine a script written with SMI's software, how are you going to figure it out?

This problem is not unique to proprietary systems (although I contend it's more likely to happen with proprietary than with open systems), and consequently the best way to future-proof your experiment is to write extensive documentation in plain text. This documentation should explain exactly what the script presents to the participants, in as much detail as possible. While writing this sounds like a chore, you're going to write (a version of) this *anyway* in the methods section of your paper.

A crucial nuance to consider when writing documentation is that the script is a set of instructions for the computer in what to do, often at a relatively low level. The documentation, on the other hand, primarily describes the participant experience. The script may direct the computer to load a sound file into the memory buffer in preparation for the next trial, so that the auditory stimulus onset is synchronous with a visual stimulus onset; the documentation would state that the stimulus onsets were simultaneous. That's not to say that the documentation *shouldn't* contain technical details—it definitely should, when appropriate!—but that the primary goal of the documentation is to allow another human researcher to understand and reproduce the experiment.

In other words, the documentation is not simply a recapitulation of the script in human-readable format. It is a high-level description of what is presented to the participant and how the responses are recorded. This high-level description can also help with cross-platform issues when trying to (re)implement the experiment on a different

computer with a different operating system. For example, the TP17 PsychoPy script was developed and tested on a machine running Debian configured to British English language settings, but the experiment was conducted on machines running Windows 7 with French language settings and a slightly different set of input hardware. Thanks to the explicit documentation, there were no issues in getting the script to work on the different machines.

6 Responses from the experiment

Finally, we get to the "real" data. Depending on your experimental presentation software, this will vary in format—for example, SuperLab and PsychoPy output text files by default, while E-Prime uses its own proprietary E-DataAid format (which can be converted to text files via E-Prime software tools).

Data files are nearly always unintelligible without documentation. The data files from TP17 have thirty-seven columns of data each, many of them with some variation of "resp" in the name. These include the following: key_resp_leftorright.keys, key_resp_leftorright.rt, key_resp_begin.keys, key_resp_begin.rt, key_resp_lexdec.keys, key_resp_lexdec.rt, key_resp_end_of_training.keys, key_resp_end_of_training.rt, key_resp_memory_test.keys, key_resp_memory_test.corr, key_resp_memory_test.rt, key_resp_memory_end.keys, and key_resp_memory_end.rt. Which one is the response to the experimental task? The first part of each column name, namely "key_resp," tells us that these columns record information about some kind of key response. The middle part of the name, for example, "lexdec" or "end_of_training," refers to the kind of trial this response was for. The column names ending in ".rt" list reaction times (i.e., the time elapsed between the presentation of the trial and the participant pressing a button), and the ones ending ".keys" show which key was pressed in response. The responses to the lexical decision task are therefore contained in key_resp_lexdec.keys, and the reaction times for these responses are in key_resp_lexdec.rt. Some other columns are less important, for example, the column key_resp_leftorright.rt tells us how quickly the participant answered the question about whether they were right-handed or left-handed.

From the perspective of the person who coded the experiment, the meaning of these column titles is relatively clear, but this system is rather opaque for anyone else. Here, again, having plain text documentation to accompany the data is key. This documentation should explain what each column represents—or at the very least, which columns can be ignored—and how to read the contents of each column.

7 Statistical analysis scripts

The first step in data analysis is data preparation. For TP17, we had a data preparation R[9] script that read each of the data files into a data frame, removed unnecessary columns (such as frame rate information) and rows (such as practice trials), calculated response accuracy, merged in the stimuli metadata (such as word frequency) and participant metadata, resulting in a data frame that was ready to be used as input to a statistical model.

The beginning of the R script contains a description of what the script does. This information carries a degree of redundancy with the script's description in this folder's readme file, although the R script description is more detailed and uses more technical language. It also states that the script is not intended to be run on its own, but instead it is called directly by the main analysis script. As before, documentation is key here.

For your own project, you need documentation. Even if you don't write a data preparation script—even if you just copy and paste each individual data file into one big Excel file (please don't do this)—write a plain text document describing what you did in more detail than you think necessary.

The other script we had for TP17 was our main analysis script. This is the one that the user actually runs to implement our analysis. The code is (somewhat) commented, explaining what the script does at both a high level (e.g., "compare the unrelated primes to the related primes") and a low level (e.g., "this function automatically re-codes factor contrasts into Helmert coding after using droplevels()").

Finally, there is a plain text readme file in this directory. This document describes each of the files in the directory, and what files to run in what order to carry out the analyses reported in the paper.

8 Storing, sharing, and citing

For TP17, due to data privacy restrictions, we are not able to share the raw data (i.e., the log files output by

the PsychoPy script) publicly. Nevertheless, here are a few considerations relating to the storage, sharing, and citation of data.

Storing data during a project is an issue everyone has to deal with. It's not uncommon for projects to have multiple collaborators, each of whom requires access to some portion of the data. A good way to achieve this while minimizing conflicting copies of data is for each collaborator to have a local copy of the data that is synchronized to a "master" copy somewhere. There are several commercial solutions such as Dropbox,[10] Google Drive,[11] Bitbucket,[12] and GitHub[13] that allow for easy cloud-based synchronization and, depending on the service, some degree of version control.[14] The cloud is just someone else's computer and is therefore sensitive to data security vulnerabilities. It's also possible to avoid using "the cloud" (thereby sidestepping possible data security vulnerabilities) via the combined use of distributed (non-cloud-based) syncing tools such as Resilio Sync and version control software such as Git or Apache Subversion, or SVN.[15]

Once the data are ready to share with the public, the question of hosting arises. Two prominent nonprofit repositories for hosting experimental data include the Tromsø Repository of Language and Linguistics (TROLLing)[16] and the Open Science Foundation (OSF).[17] Both TROLLing and OSF are able to assign a DOI (digital object identifier) to any repository, making the citation of these repositories straightforward. For more detailed discussion of these issues, see Andreassen (chapter 7, this volume) and Buszard-Welcher (chapter 10, this volume).

9 Conclusion

From this overview, an overall structure for data management emerges. Each distinct form of data should have its own directory, and each directory has an accompanying readme file explaining the contents of the directory. The readme file should also make clear the data processing workflow required—for example, describing the use of the relevant Praat script, data preparation code, or experimental procedures. Without these aids, the value of the data is significantly diminished. While in common parlance, *data* simply refers to the results of the experiment (the output of the experimental script), best practices in data management for a phonological perception experiment require the recognition that every step in the research process involves data.

Notes

1. This choice is a pragmatic one, not an ideological one. Like many linguists, my education in the linguistics of signed languages was extremely sparse, and I don't have the expertise to give the issues the discussion they deserve. I'm working to remedy this gap in my knowledge so that the next generation of linguists will have a better understanding of the world's languages in all modalities.

2. Perhaps that of my present self too, if we're being quite honest.

3. Wickham's definition of *tidy data* is essentially equivalent to third normal form in relational database management.

4. MATLAB: https://www.mathworks.com/products/matlab.html.

5. Praat: http://www.fon.hum.uva.nl/praat/.\.

6. *ASCII,* the American Standard Code for Information Exchange, is an influential (and relatively limited) character encoding developed in the 1960s. It is still widely used today and forms the basis for many Unicode encodings.

7. E-Prime: https://pstnet.com/products/e-prime/.

8. SuperLab: https://www.cedrus.com/superlab/.

9. R: https://www.r-project.org/.

10. Dropbox: https://dropbox.com.

11. Google Drive: https://www.google.com/drive.

12. Bitbucket: https://bitbucket.org.

13. GitHub: https://github.com.

14. Services that are primarily marketed for software engineering projects (such as GitHub and Bitbucket) usually have version control as a fundamental part of the system, while services such as Google Drive or Dropbox may have only rudimentary built-in version control. Note, however, that one can easily make a local Git repository and synchronize it with a Dropbox folder, thus combining the version control capabilities of Git with the ease of use of Dropbox.

15. Nothing is immune from security vulnerabilities, including distributed synchronization systems. However, it could be argued that a large system such as Google Drive is a more likely target for an organized hack attempt than a distributed synchronized system of a group of academics. Perrin (2007) termed this feature "accidental security through obscurity."

16. TROLLing: https://dataverse.no/dataverse/trolling.

17. OSF: https://osf.io.

References

Boersma, Paul, and David Weenink. 2020. Praat: Doing phonetics by computer [computer program]. Version 6.1.36. http://www.praat.org/.

Brouwer, Susanne, Holger Mitterer, and Falk Huettig. 2012. Can hearing *puter* activate *pupil*? Phonological competition and the processing of reduced spoken words in spontaneous conversations. *Quarterly Journal of Experimental Psychology* 65 (11): 2193–2220.

Cole, Jennifer, Yoonsook Mo, and Mark Hasegawa-Johnson. 2010. Signal-based and expectation-based factors in the perception of prosodic prominence. *Laboratory Phonology* 1 (2): 425–452.

Damian, Markus F., and Jeff S. Bowers. 2010. Orthographic effects in rhyme monitoring tasks: Are they automatic? *European Journal of Cognitive Psychology* 22 (1): 106–116.

Dye, Matthew W. G., and Shui-I Shih. 2006. Phonological priming in British Sign Language. In *Laboratory Phonology*, vol. 8, edited by Aditi Lahiri, 241–261. Berlin: de Gruyter Moutin.

Finley, Sara. 2011. The privileged status of locality in consonant harmony. *Journal of Memory and Language* 65:74–83.

Finley, Sara. 2012. Testing the limits of long-distance learning: Learning beyond a three-segment window. *Cognitive Science* 36:740–756.

Hay, Jennifer B., Katie Drager, and Andy Gibson. 2018. Hearing r-sandhi: The role of past experience. *Language* 94 (2): 360–404.

Hofstadter, Douglas. 1979. *Gödel, Escher, Bach: An Eternal Golden Braid*. New York: Basic Books.

Ito, Kiwako, and Shari R. Speer. 2008. Anticipatory effects of intonation: Eye movements during instructed visual search. *Journal of Memory and Language* 58:541–573. http://doi.org/:10.1016/j.jml.2007.06.013.

Katz, Jonah, and Melinda Fricke. 2018. Auditory disruption improves word segmentation: A functional basis for lenition phenomena. *Glossa* 3 (1): 38.

Ladd, D. Robert, Rory Turnbull, Charlotte Browne, Catherine Caldwell-Harris, Lesya Ganushchak, Kate Swoboda, Verity Woodfield, and Dan Dediu. 2013. Patterns of individual differences in the perception of missing-fundamental tones. *Journal of Experimental Psychology: Human Perception and Performance* 39 (5): 1386–1397.

McGurk, Harry, and John MacDonald. 1976. Hearing lips and seeing voices. *Nature* 264 (5588): 746–748.

Mitterer, Holger. 2006. On the causes of compensation for coarticulation: Evidence for phonological mediation. *Perception and Psychophysics* 68:1227–1240.

New, Boris, Christophe Pallier, Ludovic Ferrand, and Rafael Matos. 2001. Une base de données lexical du français contemporain sur internet: LEXIQUE. *L'Année Psychologique* 101:447–462.

Peirce, Jonathan, Jeremy R. Gray, Sol Simpson, Michael MacAskill, Richard Höchenberger, Hiroyuki Sogo, Erik Kastman, and Jonas Kristoffer Lindeløv. 2019. PsychoPy2: Experiments in behavior made easy. *Behavior Research Methods* 51:195–203.

Perrin, Chad. 2007. The value of accidental security through obscurity. *TechRepublic* (blog). December 13. https://www.techrepublic.com/blog/it-security/the-value-of-accidental-security-through-obscurity/.

Turnbull, Rory, and Sharon Peperkamp. 2017. The asymmetric contribution of consonants and vowels to phonological similarity: Evidence from lexical priming. *Mental Lexicon* 12 (3): 404–430.

Turnbull, Rory, Adam J. Royer, Kiwako Ito, and Shari R. Speer. 2017. Prominence perception is dependent on phonology, semantics, and awareness of discourse. *Language, Cognition and Neuroscience* 32 (8): 1017–1033.

Warren, Richard M., Keri R. Riener, James A. Bashford, and Bradley S. Brubaker. 1995. Spectral redundancy: Intelligibility of sentences heard through narrow spectral slits. *Perception and Psychophysics* 57 (2): 175–182.

Wickham, Hadley. 2014. Tidy data. *Journal of Statistical Software* 59 (10): 1–23.

Wiener, Seth, and Rory Turnbull. 2016. Constraints of tones, vowels and consonants on lexical selection in Mandarin Chinese. *Language and Speech* 59 (1): 59–82.

Youngdahl, Carla L., Eric W. Healy, Sarah E. Yoho, Frédéric Apoux, and Rachael Frush Holt. 2018. The effect of remote masking on the reception of speech by young school-age children. *Journal of Speech, Language, and Hearing Research* 61 (2): 420–427.

This book was set in Stone Serif and Stone Sans by Westchester Publishing Services.

Library of Congress Cataloging-in-Publication Data

Names: Berez-Kroeker, Andrea L., editor. | McDonnell, Bradley James, editor. | Koller, Eve, editor. | Collister, Lauren B., editor.
Title: The open handbook of linguistic data management / edited by Andrea L. Berez-Kroeker, Bradley McDonnell, Eve Koller and Lauren B. Collister.
Description: Cambridge, Massachusetts : The MIT Press, [2021] | Series: Open handbooks in linguistics series | Includes bibliographical references and index.
Identifiers: LCCN 2020044363 | ISBN 9780262045261 (hardcover)
Subjects: LCSH: Computational linguistics. | Natural language processing (Computer science) | Data mining.
Classification: LCC P98 .O64 2021 | DDC 410.285—dc23
LC record available at https://lccn.loc.gov/2020044363