Peekbank: Exploring children's word recognition through an open, large-scale repository for developmental eye-tracking data

Peekbank team, Martin Zettersten[1], Claire Bergey[2], Naiti S. Bhatt[3], Veronica Boyce[4], Mika Braginsky[5], Alexandra Carstensen[4], Benny deMayo[1], George Kachergis[4], Molly Lewis[6], Bria Long[4], Kyle MacDonald[7], Jessica Mankewitz[4], Stephan Meylan[5,8], Annissa N. Saleh[9], Rose M. Schneider[10], Angeline Sin Mei Tsui[4], Sarp Uner[8], Tian Linger Xu[11], Daniel Yurovsky[6], & Michael C. Frank[1]

[1] Dept. of Psychology, Princeton University

[2] Dept. of Psychology, University of Chicago

[3] Scripps College

[4] Dept. of Psychology, Stanford University

[5] Dept. of Brain and Cognitive Sciences, MIT

[6] Dept. of Psychology, Carnegie Mellon University

[7] Core Technology, McD Tech Labs

[8] Dept. of Psychology and Neuroscience, Duke University

[9] Dept. of Psychology, UT Austin

[10] Dept. of Psychology, UC San Diego

[11] Dept. of Psychological and Brain Sciences, Indiana University

Abstract

The ability to rapidly recognize words and link them to referents in context is central to children's early language development. This ability, often called word recognition in the developmental literature, is typically studied in the looking-while-listening paradigm, which measures infants' fixation on a target object (vs. a distractor) after hearing a target label. We present a large-scale, open database of infant and toddler eye-tracking data from looking-while-listening tasks. The goal of this effort is to address theoretical and methodological challenges in measuring vocabulary development.

Keywords: tools; processing; analysis / usage examples

Word count: X

Peekbank: Exploring children's word recognition through an open, large-scale repository for developmental eye-tracking data

Across their first years of life, children learn words at an accelerating pace (Frank, Braginsky, Yurovsky, & Marchman, 2021). Although many children will only produce their first word at around one year of age, they show signs of understanding many common nouns (e.g., "mommy") and phrases (e.g., "Let's go bye-bye!") much earlier in development (Bergelson & Swingley, 2012). However, the processes involved in early word understanding are less directly apparent in children's behaviors and are less accessible to observation than developments in speech production (Fernald, Zangl, Portillo, & Marchman, 2008). To understand speech, children must process the incoming auditory signal and link that signal to relevant meanings – a process often referred to as word recognition. Measuring early word recognition offers insight into children's early word representations and as well as the speed and efficiency with which children comprehend language in real time, as the speech signal unfolds (Bergelson, 2020; Fernald, Pinto, Swingley, Weinberg, & McRoberts, 1998). Word recognition skills are also thought to build a foundation for children's subsequent language development. Past research has found that early word recognition efficiency is predictive of later linguistic and general cognitive outcomes (Bleses, Makransky, Dale, Højen, & Ari, 2016; Marchman et al., 2018). One explanation for this relationship is that efficiency of word recognition facilitates subsequent word learning: the faster children are at processing speech, the more efficiently they can learn from the input in their environment (Fernald & Marchman, 2012).

While word recognition is a central part of children's language development, mapping the trajectory of word recognition skills has remained elusive. Studies investigating children's word recognition are typically limited in scope to experiments in individual labs involving small samples tested on a small set of items. This limitation makes it difficult to understand developmental changes in children's word knowledge at a broad scale. Peekbank provides an openly accessible database of eye-tracking data of children's word recognition, with the primary goal of facilitating the study of developmental changes in children's word knowledge and recognition speed.

The "Looking-While-Listening" Paradigm

Word recognition is traditionally studied in the "looking-while-listening" paradigm (alternatively referred to as the intermodal preferential looking procedure; Fernald et al., 2008; Hirsh-Pasek, Cauley, Golinkoff, & Gordon, 1987). In such studies, infants listen to a sentence prompting a specific referent (e.g., "Look at the dog!") while viewing two images on the screen (e.g., an image of a dog – the target image – and an image of a bird – the distractor image). Infants' word recognition is measured in terms of how quickly and accurately they fixate on the correct target image after hearing its label. Past research has used this same basic method to study a wide range of questions in language development. For example, the looking-while-listening paradigm has been used to investigate early noun knowledge, phonological representations of words, prediction during language processing, and individual differences in language development (Bergelson & Swingley, 2012; Golinkoff, Ma, Song, & Hirsh-Pasek, 2013; Lew-Williams & Fernald, 2007; Marchman et al., 2018; Swingley & Aslin, 2000).

TO DO: ALIGN CHALLENGES WITH tidybits/ use cases - computational reproducibility and teaching - item-level analyses

Measuring developmental change in word recognition

While the looking-while-listening paradigm has been highly fruitful in advancing understanding of early word knowledge, fundamental questions remain. One central question is how to accurately capture developmental change in the speed and accuracy of word recognition. There is ample evidence demonstrating that infants get faster and more accurate in word recognition over the first few years of life (e.g., Fernald et al., 1998).

However, precisely measuring developmental increases in the speed and accuracy of word recognition remains challenging due to the difficulty of distinguishing developmental changes in word recognition skill from changes in knowledge of specific words. This problem is particularly thorny in studies with young children, since the number of items that can be tested within a single session is limited and items must be selected in an age-appropriate manner (Peter et al., 2019). One way to overcome this challenge is to measure word recognition across development in a large-scale dataset with a wide range of items. A sufficiently large dataset would allow researchers to estimate developmental change in word recognition speed and accuracy while generalizing across changes related to specific words.

Developing methodological best-practices

A second question relates to evaluating methodological best practices. In particular, many fundamental analytic decisions vary substantially across studies, and different decisions may lead to researchers drawing different inferences about children's word recognition. For example, researchers vary in how they select time windows for analysis, transform the dependent measure of target fixations, and model the time course of word recognition (Csibra, Hernik, Mascaro, Tatone, & Lengyel, 2016; Fernald et al., 2008; Huang & Snedeker, 2020). This problem is made more complex by the fact that many of these decisions depend on a variety of design-related and participant-related factors (e.g., infant age). Establishing best practices therefore requires a large database of infant word recognition studies varying across such factors, in order to test the potential consequences of methodological decisions on study results.

Peekbank: An open database of developmental eye-tracking studies.

What these two questions share is that they are difficult to answer at the scale of a single study. To address this challenge, we introduce Peekbank, a flexible and reproducible interface to an open database of developmental eye-tracking studies. The Peekbank project (a) collects a large set of eye-tracking datasets on children's word recognition, (b) introduces a data format and processing tools for standardizing eye-tracking data across data sources, and (c) provides an interface for accessing and analyzing the database. In the current paper, we give an overview of the key components of the project and some initial demonstrations of its utility in advancing theoretical and methodological insights. We report two analyses using the database and associated tools (N=1,233): (1) a growth curve analysis modeling age-related changes in infants' word recognition while generalizing across item-level variability; and (2) a multiverse-style analysis of how a central methodological decision – selecting the time window of analysis – impacts inter-item reliability.

Design and Technical Approach

Database Framework

One of the main challenges in compiling a large-scale eye-tracking dataset is the lack of a shared data format across individual experiments. Researcher conventions for structuring data vary, as do the technical specifications of different devices (e.g., computer displays and eyetracking cameras), rendering the task of integrating datasets from different labs and data sources difficult. Therefore, our first effort was to develop a common tabular format to support analyses of all studies simultaneously.

As illustrated in Figure 1, the Peekbank framework consists of four main components: (1) a set of tools to convert eye-tracking datasets into a unified format, (2) a relational database populated with data in this unified format, (3) a set of tools to retrieve data from this database, and (4) a web app (using the Shiny framework) for visualizing the data. These components are supported by three libraries. The `peekds` library (for the R language; R Core Team (2020)) helps researchers convert existing datasets to use the standardized format of the database. The `peekbank` module (Python) creates a database with the relational schema and populates it with the standardized datasets produced by `peekds`. The database is implemented in MySQL, an industry standard relational database, which may be accessed by a variety of programming languages, and can be hosted on one machine and accessed by many others over the Internet. The `peekbankr` library (R) provides an application programming interface, or API, that offers high-level abstractions for accessing the tabular data stored in Peekbank. Most users will access data through this final library, in which case the details of data formatting and processing are abstracted away from the user.

In the following sections, we will begin by providing the details on the database's organization (or *schema*) and the technical implementation on `peekds`. Users who are primarily interested in accessing the database can skip these details and focus on access through the `peekbankr` API and the web apps.

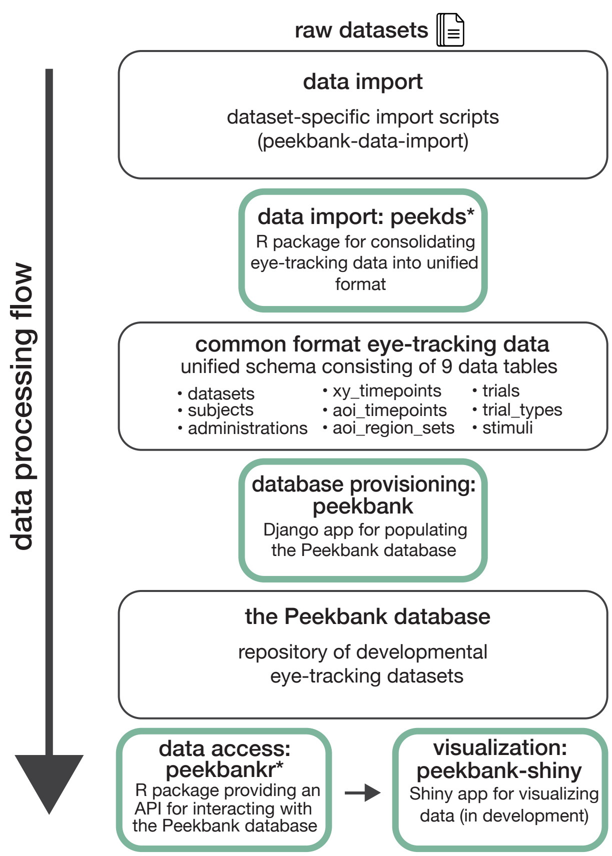

Figure 1. Overview of the Peekbank data ecosystem. Peekbank tools are highlighted in green. * indicates R packages introduced in this work.

Database Schema

The peekbank database contains two major types of data: (1) timecourse looking data, detailing where on the screen a child is looking at a given point in time, and (2) metadata regarding the relevant experiment, participant, and trial (Fig. 2). Here, we will give an outline of the tables encoding this data. As is common in relational databases, records of similar types (e.g., participants, trials, experiments, coded looks at each timepoint) are grouped into tables, and records of various types are linked through numeric identifiers.

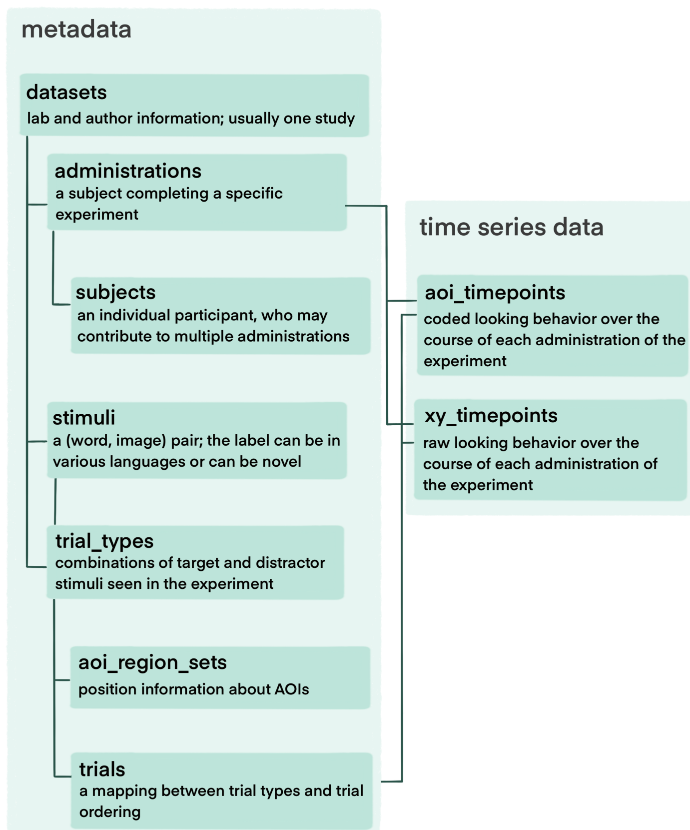

Figure 2. The Peekbank schema. Each square represents a table in the relational database.

Timecourse data. Timecourse looking data is encoded in two tables: `aoi_timepoints` and `xy_timepoints`. The `aoi_timepoints` table encodes where a child is looking at each point in time, by specifying the coded area of interest (AOI): looks to the target, looks to the distractor, looks on the screen but away from target and distractor, and missing looks. All datasets must include this timecourse data, as it represents the main record of children's looking behavior. For eyetracking experiments that are automatically rather than manually coded, the `xy_timepoints` table additionally encodes the inferred (x, y) coordinates of fixations on the screen over the course of each trial. Both the `aoi_timepoints` and `xy_timepoints` tables are resampled to a consistent sampling rate, as described in the Import section below. To normalize across trials and across experiments, all timecourses are computed so that the time of 0 ms represents the onset of disambiguating material (i.e., the beginning of *dog* in "Can you find the *dog*?").

Metadata. Each record in the timecourse data is linked to several metadata records. This metadata can be separated into three parts: (1) subject-level information (e.g., demographics) (2) experiment-level information (e.g., a subject's age for a specific experiment, or the particular eyetracker used to collect the data) and (3) trial information and experimental design (what images or videos were presented onscreen, and paired with which audio). Information about individuals who participate in one more studies, for example a subject's sex and first language, is recorded in the `subjects` table, while the `administrations` table contains information about a specific subject participating in a specific experiment. This division allows Peekbank to gracefully handle longitudinal designs: a single subject can be associated with many administrations.

The `stimuli` and `trial_types` tables store information about trials, which in turn may reflect specifics of the experiment design. Stimuli are (label, image) mappings that are seen in the experiment. The `trial_types` table encodes information about each trial of the experiment, including the target stimulus and location, the distractor stimulus and location, and the point of disambiguation for that trial. If this dataset used automatic eyetracking rather than manual coding, each trial type is additionally linked to a set of area of interest (x, y) coordinates, encoded in the `aoi_region_sets` table.

Because individual trial types can be repeated multiple times within an administration, the order of the trials is encoded in the `trials` table. Each unique ordering that occurred in the experiment is encoded in this table. The `trial_id`, which links a trial type to the order it was presented in an administration, is attached to the timecourse looking data.

Import

During data import, raw eye-tracking datasets are processed to conform to the Peekbank data schema. The following section is a description of the import process for Peekbank. It serves as both a description of our method in importing the datasets already in the database, as well as a high-level overview of the import process for researchers looking to import their data in the future. First, we will describe the import of metadata, and second, we will describe import of the timecourse looking data, including processing functions in `peekds` for normalizing and resampling looking behavior.

Metadata. Subject-level data is imported for all participants who have experiment data. In general, we import data without particular exclusions, including as many participants as possible in the database. The `subjects` and `administrations` tables separate information at the subject level from information about runs of the experiment, such that longitudinal studies have multiple administrations linked to each subject.

The `stimuli` table has a row for each (word, image) pair, and thus is used slightly differently across different experiment designs. In most experiments, there is a one-to-one mapping between images and labels (e.g., each time an image of a dog appears it is referred to as "dog"). For studies in which there are multiple potential labels per image (e.g., "dog" and "chien" are both used to refer to an image of a dog), images can have multiple rows in the `stimuli` table with unique labels as well as a row with no label to be used when the image appears solely as a distractor (and thus its label is ambiguous). This structure is useful for studies on synonymy or using multiple languages. For studies in which the same label refers to multiple images (e.g., the word "dog" refers to an image of a dalmatian and a poodle), the same label can have multiple rows in the `stimuli` table with unique images. The `trial_types` table contains each pair of stimuli, a target and distractor, seen in the experiment. The `trial_types` table links trial types to the `aoi_region_sets` table and the `trials` table.

The `trials` table encodes each unique ordering of trial types seen in all runs of an experiment. For example, for experiments with a fixed trial order, the `trials` table will have as many rows as there are stimuli in the experiment; for experiments with a randomized trial order, there will be many rows linking the trial orderings to the trial types. The `trials` table links all experiment design information to the timecourse data.

Timecourse data. Raw looking data is a series of looks to AOIs or to (x, y) coordinates on the experiment screen, linked to points in time. For data generated by eyetrackers, we typically have (x, y) coordinates at each time point, which will be encoded in the `xy_timepoints` table. These looks will also be recoded into AOIs according to the AOI coordinates in the `aoi_region_sets` table using the `add_aois()` function in `peekds`, which will be encoded in the `aoi_timepoints` table. For hand-coded data, we typically have a series of AOIs; these will be recoded into the categories in the Peekbank schema (target, distractor, other, and missing) and encoded in the `aoi_timepoints` table, and these datasets will not have an `xy_timepoints` table.

Typically, timepoints in the `xy_timepoints` table and `aoi_timepoints` table need to be regularized to center each trial's time around the point of disambiguation–the time of target word onset in the trial. If time values run throughout the experiment rather than resetting to zero at the beginning of each trial, `rezero_times()` is used to reset the time at each trial. After this, each trial's times are centered around the point of disambiguation using `normalize_times()`. When these steps are complete, the time course is ready for resampling.

To facilitate time course analysis and visualization across datasets, timecourse data must be resampled to a uniform sampling rate (i.e., such that every trial in every dataset has observations at the same time points). To do this, we use the `resample()` function. During the resampling process, we interpolate using constant interpolation, selecting for each interpolated timepoint the looking location for the nearest observed time point in the original data for both `aoi_timepoints` and `xy_timepoints` data. Compared to linear interpolation (see e.g. Wass et al., 2014), constant interpolation has the advantage that it does not introduce new look locations, so it is a more conservative method of resampling.

Validation and ingestion into the database

After resampling, the final step of dataset import is validation. The `peekds` package offers functions to check the now processed data tables against the database schema to ensure that all tables have the required fields and correct data types for database ingestion. In an effort to double check the data quality and to make sure that no errors are made in the importing script, as part of the import procedure we create a timecourse plot based on our processed tables to replicate the results in the original paper. Once this plot has been created and checked for consistency and all tables pass our validation functions, the processed dataset is ready for ingestion into the database.

Currently, the import process is carried out by the Peekbank team using data offered by other research teams. In the future, we hope to allow research teams to carry out their own import processes with checks from the Peekbank team before ingestion. To this end, import script templates are available for both hand-coded datasets and automatic eyetracking datasets for research teams to adapt to their data.

CHECK and edit resampling section for ties, interpolating forward/back in time, and for maximum time over which we interpolate

Current Data Sources

The database currently includes 11 looking-while-listening datasets comprising N=1320 total participants (Table 1). Most datasets (10 out of 11 total) consist of data from monolingual native English speakers. They span a wide age spectrum with participants

Table 1
Overview of the datasets in the current database.

Dataset name	Citation	N	Mean age (mos.)	Age range (mos.)	Method	Language
attword	Yurovsky & Frank, 2017	288	25.5	13–59	eye-tracking	English
canine	unpublished	36	23.8	21–27	manual coding	English
coartic	Mahr et al., 2015	29	20.8	18–24	eye-tracking	English
cowpig	Perry et al., 2017	45	20.5	19–22	manual coding	English
ft_pt	Adams et al., 2018	69	17.1	13–20	manual coding	English
mispron	Swingley & Aslin, 2002	50	15.1	14–16	manual coding	English
mix	Byers-Heinlein et al., 2017	48	20.1	19–21	eye-tracking	English, French
reflook_socword	Yurovsky et al., 2013	435	33.6	12–70	eye-tracking	English
reflook_v4	unpublished	45	34.2	11–60	eye-tracking	English
remix	Potter et al., 2019	44	22.6	18–29	manual coding	Spanish, English
salientme	Pomper & Saffran, 2019	44	40.1	38–43	manual coding	English
switchingCues	Pomper & Saffran, 2016	60	44.3	41–47	manual coding	English
tablet	Frank et al., 2016	69	35.5	12–60	eye-tracking	English
tseltal	Casillas et al., 2017	23	31.3	9–48	manual coding	Tseltal
yoursmy	Garrison et al., 2020	35	14.5	12–18	eye-tracking	English

ranging from 8 to 84 months of age, and are balanced in terms of gender (48% female). The datasets vary across a number of dimensions related to design and methodology, and include studies using manually coded video recordings and automated eye-tracking methods (e.g., Tobii, EyeLink) to measure gaze behavior. Most studies focused on testing familiar items, but the database also includes studies with novel pseudowords. All data (and accompanying references) are openly available on the Open Science Framework (osf.io/pr6wu).

How selected? Language coverage? More details about lab and design variation?

Versioning + Expanding the database

The content of Peekbank will change as we add additional datasets and revise previous ones. To facilitate reproducibility of analyses, we use a versioning system where successive releases are assigned a name reflecting the year and version, e.g., `2021.1`. By default, users will interact with the most recent version of the database available, though `peekbankr` API allows researchers to run analyses against any previous version of the database. For users with intensive use-cases, each version of the database may be downloaded as a compressed .sql file and installed on a local MySQL server.

Interfacing with peekbank

Shiny App

One goal of the Peekbank project is to allow a wide range of users to easily explore and learn from the database. We therefore have created an interactive web application – `peekbank-shiny` – that allows users to quickly and easily create informative visualizations of individual datasets and aggregated data. `peekbank-shiny` is built using Shiny, a software package for creating web apps using R. The Shiny app allows users to create commonly used visualizations of looking-while-listening data, based on data from the Peekbank database. Specifically, users can visualize

1. the time course of looking data in a profile plot depicting infant target looking across trial time
2. overall accuracy (proportion target looking) within a specified analysis window
3. reaction times (speed of fixating the target image) in response to a target label
4. an onset-contingent plot, which shows the time course of participant looking as a function of their look location at the onset of the target label

Users are given various customization options for each of these visualizations, e.g., choosing which datasets to include in the plots, controlling the age range of participants, splitting the visualizations by age bins, and controlling the analysis window for time course analyses. Plots are then updated in real time to reflect users' customization choices, and users are given options to share the visualizations they created. The Shiny app thus allows users to quickly inspect basic properties of Peekbanks datasets and create reproducible visualizations without incurring any of the technical overhead required to access the database through R.

Peekbankr

The `peekbankr` API offers a way for users to access data from the database and flexibly analyze it in `R`. Users can download tables from the database, as specified in the Schema section above, and merge them using their linked IDs to examine timecourse data and metadata jointly. In the sections below, we work through some examples to outline the possibilities for analyzing data downloaded using `peekbankr`.

Functions:

- `connect_to_peekbank()`
- `get_datasets()`
- `get_subjects()`
- `get_administrations()`
- `get_stimuli()`
- `get_aoi_timepoints()`
- `get_trials()`
- `get_trial_types()`
- `get_xy_timepoints()`
- `get_aoi_region_sets()`

OSF site

Stimuli Data in raw format (if some additional datum needed, e.g. pupil size?)

Peekbank in Action

We provide two potential use-cases for Peekbank data. In each case, we provide sample code so as to model how easy it is to do simple analyses using data from the database. Our first example shows how we can replicate the analysis for a classic study. This type of computational reproducibility can be a very useful exercise for teaching students about best practices for data analysis (e.g., Hardwicke et al., 2018) and also provides an easy way to explore looking-while-listening timecourse data in a standardized format. Our second example shows an in-depth exploration of developmental changes in the recognition of particular words. Besides its theoretical interest (which we will explore more fully in subsequent work), this type of analysis could in principle be used for optimizing the stimuli for new experiments, especially as the Peekbank dataset grows and gains coverage over a great number of items.

Computational reproducibility example: Swingley and Aslin (2000)

Swingley and Aslin (2000) investigated the specificity of 14-16 month-olds' word representations using the looking-while-listening paradigm, asking whether recognition would be slower and less accurate for mispronunciations, e.g. "oppel" (close mispronunciation) or "opel" (distant mispronunciation) instead of "apple" (correct condition). In this short vignette, we show how easily the data in Peekbank can be used to visualize this result.

```
library(peekbankr)
aoi_timepoints <- get_aoi_timepoints(dataset_name = "swingley_aslin_2002")
administrations <- get_administrations(dataset_name = "swingley_aslin_2002")
trial_types <- get_trial_types(dataset_name = "swingley_aslin_2002")
trials <- get_trials(dataset_name = "swingley_aslin_2002")
```

We begin by retrieving the relevant tables from the database, `aoi_timepoints`, `administrations`, `trial_types`, and `trials`. As discussed above, each of these can be downloaded using a simple API call through `peekbankr`, which returns dataframes that include ID fields. These ID fields allow for easy joining of the data into a single dataframe containing all the information necessary for the analysis.

```
swingley_data <- aoi_timepoints %>%
  left_join(administrations) %>%
  left_join(trials) %>%
  left_join(trial_types) %>%
  filter(condition != "filler") %>%
```

```
  mutate(condition = if_else(condition == "cp", "Correct", "Mispronounced"))
```

As the code above shows, once the data are joined, condition information for each timepoint is present and so we can easily filter out filler trials and set up the conditions for further analysis. For simplicity, here we combine both mispronunciation conditions since this manipulation showed no effect in the original paper.

```
accuracies <- swingley_data  %>%
  group_by(condition, t_norm, administration_id) %>%
  summarize(correct = sum(aoi == "target") /
              sum(aoi %in% c("target","distractor"))) %>%
  group_by(condition, t_norm) %>%
  summarize(mean_correct = mean(correct),
            ci = 1.96 * sd(correct) / sqrt(n()))
```

The final step in our analysis is to create a summary dataframe using `dplyr` commands. We first group the data by timestep, participant, and condition and compute the proportion looking at the correct image. We then summarize again, averaging across participants, computing both means and 95% confidence intervals (via the approximation of 1.96 times the standard error of the mean). The resulting dataframe can be used for visualization of the time-course of looking.

```
ggplot(accuracies, aes(x = t_norm, y = mean_correct, color = condition)) +
  geom_hline(yintercept = 0.5, linetype = "dashed", color = "black") +
  geom_vline(xintercept = 0, linetype = "dotted", color = "black") +
  geom_pointrange(aes(ymin = mean_correct - ci,
                      ymax = mean_correct + ci)) +
  labs(x = "Time from target word onset (msec)",
       y = "Proportion looking at correct image",
       color = "Condition") +
  lims(x = c(-500, 3000))
```

Figure 3 shows the average time course of looking for the two conditions, as produced by the code above. Looks after the correctly pronounced noun appeared both faster (deviating from chance earlier) and more accurate (showing a higher asymptote). Overall, this example demonstrates the ability to produce this visualization in just a few lines of code.

Item analyses

A second use case for Peekbank is to examine item-level variation in word recognition. Individual datasets rarely have enough statistical power to show reliable developmental differences within items. To illustrate the power of aggregating data across multiple datasets, we select the four words with the most data available across studies and ages (apple, book, dog, and frog) and show average recognition trajectories.

Our first step is to collect and join the data from the relevant tables including timepoint data, trial and stimulus data, and administration data (for participant ages). We join these into a single dataframe for easy manipulation; this dataframe is a common starting point for analyses of item-level data.

```r
all_aoi_timepoints <- get_aoi_timepoints()
all_stimuli <- get_stimuli()
all_administrations <- get_administrations()
all_trial_types <- get_trial_types()
all_trials <- get_trials()

aoi_data_joined <- all_aoi_timepoints %>%
  right_join(all_administrations) %>%
  right_join(all_trials) %>%
  right_join(all_trial_types) %>%
  mutate(stimulus_id = target_id) %>%
  right_join(all_stimuli) %>%
  select(administration_id, english_stimulus_label, age, t_norm, aoi)
```

Next we select a set of four target words (chosen based on having more than XXX children contributing data for each across several one-year age groups). We create age groups, aggregate, and compute timepoint-by-timepoint confidence intervals using the z approximation.

```
target_words <- c("book","dog","frog","apple")


target_word_data <- aoi_data_joined %>%
  filter(english_stimulus_label %in% target_words) %>%
  mutate(age_group = cut(age, breaks = seq(12,48,12))) %>%
  filter(!is.na(age_group)) %>%
  group_by(t_norm, administration_id, age_group, english_stimulus_label) %>%
  summarise(correct = mean(aoi == "target") /
              mean(aoi %in% c("target","distractor"), na.rm=TRUE)) %>%
  group_by(t_norm, age_group, english_stimulus_label) %>%
  summarise(ci = 1.96 * sd(correct, na.rm=TRUE) / sqrt(length(correct)),
            correct = mean(correct, na.rm=TRUE),
            n = n())
```

Finally, we plot the data as timecourses split by age. Our plotting code is shown below (with styling commands again removed for clarity). Figure 4 shows the resulting plot, with time courses for each of three (rather coarse) age bins. Although some baseline effects are visible across items, we still see clear and consistent increases in looking to the target, with the increase appearing earlier and in many cases asymptoting at a higher level for older children. On the other hand, this simple averaging approach ignores study-to-study variation (perhaps responsible for the baseline effects we see in the "apple" and "frog" items especially). In future work, we hope to introduce model-based analytic methods that use mixed effects regression to factor out study-level and individual-level variance in order to recover developmental effects more appropriately (see e.g. Zettersten et al. (2021) for a prototype of such an analysis).

```
ggplot(target_word_data,
       aes(x = t_norm, y = correct, col = age_group)) +
  geom_line() +
  geom_linerange(aes(ymin = correct - ci, ymax = correct + ci),
                 alpha = .2) +
  facet_wrap(~english_stimulus_label)
```

Discussion and Conclusion

Theoretical progress in understanding child development requires rich datasets, but collecting child data is expensive, difficult, and time-intensive. Recent years have seen a growing effort to build open source tools and pool research efforts to meet the challenge of building a cumulative developmental science (Bergmann et al. (2018); Frank, Braginsky, Yurovsky, and Marchman (2017); The ManyBabies Consortium (2020)]. The Peekbank project expands on these efforts by building an infrastructure for aggregating eye-tracking data across studies, with a specific focus on the looking-while-listening paradigm. This paper presents an illustration of some of the key theoretical and methodological questions that can be addressed using Peekbank: generalizing across item-level variability in children's word recognition and providing data-driven guidance on methodological choices.

There are a number of limitations surrounding the current scope of the database. A priority in future work will be to expand the size of the database. With 11 datasets currently available in the database, idiosyncrasies of particular designs and condition manipulations still have substantial influence on modeling results. Expanding the set of distinct datasets will allow us to increase the number of observations per item across datasets, leading to more robust generalizations across item-level variability. The current database is also limited by the relatively homogeneous background of its participants, both with respect to language (almost entirely monolingual native English speakers) and cultural background (all but one dataset come from WEIRD populations, potentially limiting generalizability; see Muthukrishna et al. (2020)). Increasing the diversity of participant backgrounds and languages will expand the scope of the generalizations we can form about child word recognition.

Finally, while the current database is focused on studies of word recognition, the tools and infrastructure developed in the project can in principle be used to accommodate any eye-tracking paradigm, opening up new avenues for insights into cognitive development. Gaze behavior has been at the core of many of the key advances in our understanding of infant cognition. Aggregating large datasets of infant looking behavior in a single, openly-accessible format promises to bring a fuller picture of infant cognitive development into view.

Acknowledgements

We would like to thank the labs and researchers that have made their data publicly available in the database.

References

Bergelson, E. (2020). The comprehension boost in early word learning: Older infants are better learners. *Child Development Perspectives*, *14*(3), 142–149.

Bergelson, E., & Swingley, D. (2012). At 6-9 months, human infants know the meanings of many common nouns. *PNAS*, *109*(9), 3253–3258.

Bergmann, C., Tsuji, S., Piccinini, P. E., Lewis, M. L., Braginsky, M., Frank, M. C., & Cristia, A. (2018). Promoting replicability in developmental research through meta-analyses: Insights from language acquisition research. *Child Development*, *89*(6), 1996–2009.

Bleses, D., Makransky, G., Dale, P. S., Højen, A., & Ari, B. A. (2016). Early productive vocabulary predicts academic achievement 10 years later. *Applied Psycholinguistics*, *37*(6), 1461–1476.

Csibra, G., Hernik, M., Mascaro, O., Tatone, D., & Lengyel, M. (2016). Statistical treatment of looking-time data. *Developmental Psychology*, *52*(4), 521–536.

Fernald, A., & Marchman, V. A. (2012). Individual differences in lexical processing at 18 months predict vocabulary growth in typically developing and late-talking toddlers. *Child Development*, *83*(1), 203–222.

Fernald, A., Pinto, J. P., Swingley, D., Weinberg, A., & McRoberts, G. W. (1998). Rapid gains in speed of verbal processing by infants in the 2nd year. *Psychological Science*, *9*(3), 228–231.

Fernald, A., Zangl, R., Portillo, A. L., & Marchman, V. A. (2008). Looking while listening: Using eye movements to monitor spoken language comprehension by infants and young children. In I. A. Sekerina, E. M. Fernandez, & H. Clahsen

(Eds.), *Developmental psycholinguistics: On-line methods in children's language processing* (pp. 97–135). Amsterdam: John Benjamins.

Frank, M. C., Braginsky, M., Yurovsky, D., & Marchman, V. A. (2017). Wordbank: An open repository for developmental vocabulary data. *Journal of Child Language*, *44*(3), 677–694.

Frank, M. C., Braginsky, M., Yurovsky, D., & Marchman, V. A. (2021). *Variability and Consistency in Early Language Learning: The Wordbank Project*. Cambridge, MA: MIT Press.

Golinkoff, R. M., Ma, W., Song, L., & Hirsh-Pasek, K. (2013). Twenty-five years using the intermodal preferential looking paradigm to study language acquisition: What have we learned? *Perspectives on Psychol. Science*, *8*(3), 316–339.

Hardwicke, T. E., Mathur, M. B., MacDonald, K., Nilsonne, G., Banks, G. C., Kidwell, M. C., ... Frank, M. C. (2018). Data availability, reusability, and analytic reproducibility: Evaluating the impact of a mandatory open data policy at the journal Cognition. *Royal Society Open Science*, *5*(8). https://doi.org/10.1098/rsos.180448

Hirsh-Pasek, K., Cauley, K. M., Golinkoff, R. M., & Gordon, L. (1987). The eyes have it: Lexical and syntactic comprehension in a new paradigm. *Journal of Child Language*, *14*(1), 23–45.

Huang, Y., & Snedeker, J. (2020). Evidence from the visual world paradigm raises questions about unaccusativity and growth curve analyses. *Cognition*, *200*, 104251.

Lew-Williams, C., & Fernald, A. (2007). Young children learning Spanish make rapid use of grammatical gender in spoken word recognition. *Psychological Science*, *18*(3), 193–198.

Marchman, V. A., Loi, E. C., Adams, K. A., Ashland, M., Fernald, A., & Feldman, H. M. (2018). Speed of language comprehension at 18 months old predicts school-relevant outcomes at 54 months old in children born preterm. *Journal of Dev. & Behav. Pediatrics*, *39*(3), 246–253.

Muthukrishna, M., Bell, A. V., Henrich, J., Curtin, C. M., Gedranovich, A., McInerney, J., & Thue, B. (2020). Beyond Western, Educated, Industrial, Rich, and Democratic (WEIRD) Psychology: Measuring and Mapping Scales of Cultural and Psychological Distance. *Psychological Science*, *31*(6), 678–701.

Peter, M. S., Durrant, S., Jessop, A., Bidgood, A., Pine, J. M., & Rowland, C. F. (2019). Does speed of processing or vocabulary size predict later language growth in toddlers? *Cognitive Psychology*, *115*, 101238.

R Core Team. (2020). *R: A language and environment for statistical computing.* Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

Swingley, D., & Aslin, R. N. (2000). Spoken word recognition and lexical representation in very young children. *Cognition*, *76*(2), 147–166.

The ManyBabies Consortium. (2020). Quantifying sources of variability in infancy research using the infant-directed speech preference. *Advances in Methods and Practices in Psychological Science*, *3*(1), 24–52.

Zettersten, M., Bergey, C., Bhatt, N., Boyce, V., Braginsky, M., Carstensen, A., … others. (2021). Peekbank: Exploring children's word recognition through an open, large-scale repository for developmental eye-tracking data.

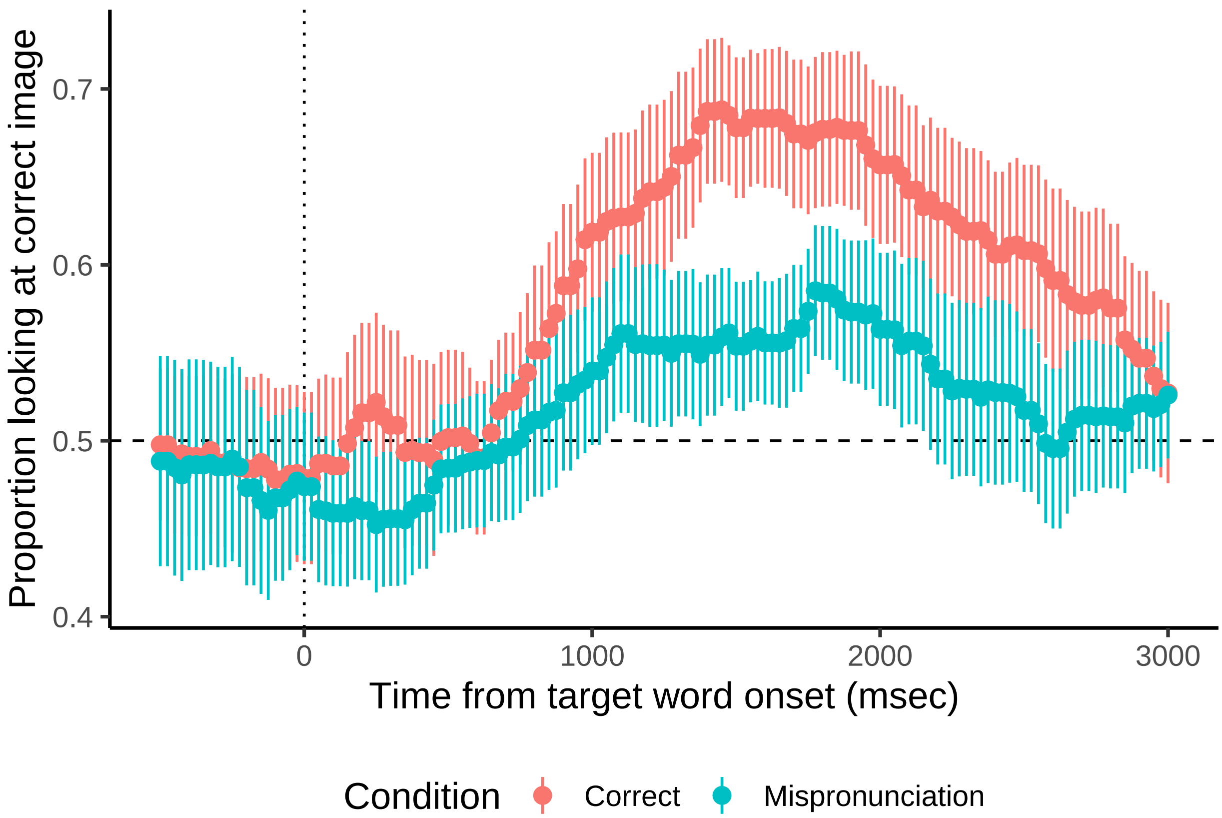

Figure 3. Proportion looking at the correct referent by time from the point of disambiguation (the onset of the target noun). Colors show the two pronunciation conditions; points give means and ranges show 95% confidence intervals. The dotted line shows the point of disambiguation and the dashed line shows chance performance.

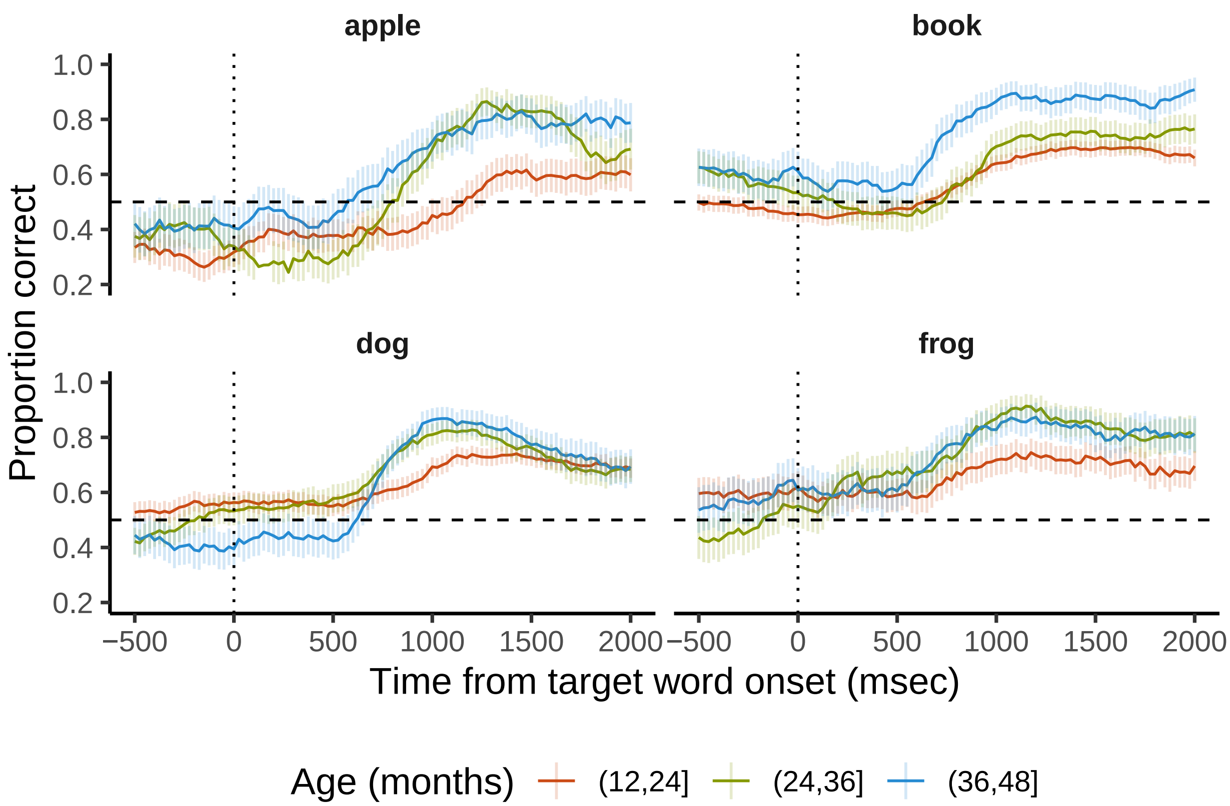

Figure 4. Add caption here.