A-maze of Natural Stories: Comprehension and surprisal in the Maze task

Veronica Boyce[1] & Roger Levy[2]

[1] Stanford University
[2] Massachusetts Institute of Technology

Author Note

We thank the AMLAP 2020 audience, the Computational Psycholinguistics Lab at MIT, the Language and Cognition Lab at Stanford, the QuantLang Lab at UC Irvine, and Mike Frank for feedback on this work.

The authors made the following contributions. Veronica Boyce: Conceptualization, Formal Analysis, Investigation, Software, Writing - Original Draft Preparation, Writing - Review & Editing; Roger Levy: Conceptualization, Formal Analysis, Writing - Review & Editing.

Correspondence concerning this article should be addressed to Veronica Boyce, 450 Jane Stanford Way, Building 420, Stanford University, Stanford, CA 94305. E-mail: vboyce@stanford.edu

Abstract

Behavioral measures of word-by-word reading time provide experimental evidence to test theories of language processing. A-maze is a recent method for measuring incremental sentence processing that can localize slowdowns related to syntactic ambiguities in individual sentences. We adapted A-maze for use on longer passages and tested it on the Natural Stories corpus. Participants were able to comprehend these longer text passages that they read via the Maze task. Moreover, the Maze task yielded useable reaction time data with word predictability effects that were linearly related to surprisal, the same pattern found with other incremental methods. Crucially, Maze reaction times show a tight relationship with properties of the current word, but little spillover of effects from previous words. This superior localization is an advantage of Maze compared with other methods. Overall, we expanded the scope of experimental materials, and thus theoretical questions, that can be studied with the Maze task.

A-maze of Natural Stories: Comprehension and surprisal in the Maze task

Introduction

Behavioral measures of word-by-word reading time provide experimental evidence to test theories of language processing. Maze is a potentially promising method, and it's feasibility has recently increased due to the introduction of A-Maze (Boyce, Futrell, & Levy, 2020). We tested A-maze on a naturalistic corpora to learn whether participants could understand stories read in the demanding Maze task and whether the reaction time profiles from the Maze task matched those from other methods. To provide context, we first lay out the role of incremental processing methods in theory testing, describe a few common methods, and review some key influences on reading time.

Why measure reading time?

Occasionally we read an unexpected word, cannot make sense of it and have to read the sentence again. The words that cause these stumbles are the outliers on a distribution of words varying in how easy they are to process in context. Even when we don't notice a problem, less expected words still take longer to process as we (re)construct a mental model of the sentence. Fortunately for reading and unfortunately for studying language, this process of adjustment is very quick, with millisecond scale differences in reading time between words.

If we assume that processing time reflects something about the processing, be it character recognition, memory retrieval, parsing, or something else, then we can learn something about how the mind processes language from measuring how long the processing takes in different circumstances. With this assumption, psycholinguists can test theories that predict the relative difficulty of two constructions by measuring the relative processing time of these words in context.

For instance, theories about what makes object relative clauses harder to understand than subject relative clauses make different predictions about which words are the loci of the overall difficulty and slower reading times associated with object relatives (Grodner & Gibson, 2005; Staub, 2010; Traxler, Morris, & Seely, 2002). Measures of reading time can also inform theories about the time course of processing (i.e. which steps are parallel versus serial, Bartek, Lewis, Vasishth, and Smith (2011)) or the functional form of relationships between word characteristics and processing time (Smith & Levy, 2013).

Some of these theories rely on being able to attribute processing slowdowns to a particular word. Determining that object relatives are overall slower that subject relatives is easy. Even an imprecise measure of reading time will determine that the same set of words in a different order took longer to read at a sentence level. However, many language processing theories make specific (and contrasting) theories about which words in a sentence are harder to process. To adjudicate these theories, we want methods that are *localized*, so it is easy to determine which word is responsible for an observed slow-down in reading time. Ideally, a longer reading time on a word would be an indication of that word's increased difficulty, and not the lingering signal of a prior word's increased difficulty. When the signal isn't localized, advanced analysis techniques may be required to disentangle the slow-downs (Shain & Schuler, 2018).

Eye-tracking and Self-paced reading

Behavioral methods that measure how long it takes to read each word in a sentence are referred to as incremental processing methods, and their dependent measures are called reading times or reaction times, both abbreviated RT. The two most commonly used methods are eye-tracking and self-paced reading, but both of these suffer from a lack of localization.

In eye-tracking, participants read a text on a screen naturally, while their eye-movements are recorded (Rayner, 1998). The downside of natural reading is that people often skip short words and look ahead or look back as they read, the dynamics of which make it hard to isolate effects (Frazier & Rayner, 1982; Levy, Bicknell, Slattery, & Rayner, 2009; Rayner, Ashby, Pollatsek, & Reichle, 2004).

Self-paced reading (SPR) is a more controlled process where a participant sees one word on screen at a time and presses a button to see the next word instead. Even in SPR, readers may maintain ambiguities about what a word was or means until it is later resolved by context, so it may take multiple words for slowdowns to catch up with them (Koornneef & van Berkum, 2006; MacDonald, 1993).

Both these methods are prone to spillover. The time to plan motor movements like saccades or button presses is substantial compared to the time it takes to recognize a word, these movements are often initiated before a word is fully processed. By the time any late processing difficulties occur, the reader may already be several words along.

Maze

The Maze task is an alternative method that is designed to increase localization at the expense of naturalness (Forster, Guerrera, & Elliot, 2009; Freedman & Forster, 1985). In the Maze task, participants see two words at a time: a correct word that continues the sentence, and a distractor which does not. Participants must choose the correct word, and their time to selection is treated as the reaction time (RT). Forster et al. (2009) introduced two versions of the Maze task: lexical L-maze where the distractors are nonce words and grammatical G-maze where the distractors are real words that don't fit with the context of the sentence. Theoretically, participants must fully integrate each word into the sentence in order to confidently select it, which may require mentally reparsing previous material in order to allow the integration and selection of a disambiguating word. Forster et al. (2009) call this need for full integration "forced incremental processing" to distinguish from other incremental processing methods where words can be passively read before later committing to a parse. This idea of strong localization is supported by studies finding strongly localized effects for G-maze (Boyce et al., 2020; Witzel, Witzel, & Forster, 2012).

The downside of G-maze is that materials are effort-intensive to construct because of the need to select infelicitous words as distractors for each spot of each sentence. This burdensome preparation may explain why the Maze task has not been widely adopted. Boyce et al. (2020) demonstrated a way to automatically generate Maze distractors by using language models from Natural Language Processing to find words that are high surprisal in the context of the target sentence, and thus likely to be judged infelicitous by human readers. Boyce et al. (2020) call Maze with automatically generated distractors A-maze. In a comparison, A-maze distractors had similar results to the hand-generated G-maze distractors from Witzel et al. (2012) and A-maze outperformed L-maze and an SPR control in detecting and localizing expected slowdown effects. Sloggett, Handel, and Rysling (2020) also found that A-maze and G-maze distractors yielded similar results on a disambiguation paradigm.

Another recent variant of the Maze task is interpolated I-maze, which uses a mix of real word distractors (generated via the A-maze process) and nonce word distractors (Vani, Wilcox, & Levy, 2021; Wilcox, Vani, & Levy, 2021). The presence of real word distractors encourages close attention to the sentential context, while nonce words can be used as distractors where the word in the sentence is itself ungrammatical or highly unexpected.

Measuring localization: Frequency, length, and surprisal effects

Localized measures can be used to attribute processing difficulty to individual words; however, to determine if a method is localized requires knowing how hard the words were to process. One approach is to look at properties of words that are known to influence reading times across methods such as eye-tracking and SPR. Longer words and lower frequency words tend to take longer to process (Kliegl, Grabner, Rolfs, & Engbert, 2004), as do less predictable words (Rayner et al., 2004).

A word can be unpredictable for a variety of reasons: it could be low frequency, semantically unexpected, the start of a low-frequency syntactic construction, or a word that disambiguates prior words to a less common parse. Many targeted effects of interest are essentially looking at specific features that contribute to how predictable or unpredictable a word is. Thus incremental processing methods that are sensitive to predictability are useful for testing linguistic theories that make predictions about what words are unexpected.

The overall predictability of a word in a context can be estimated using language models that are trained on large corpora of language to predict what word comes next in a sentence. A variety of pre-trained models exist, with varied internal architectures and training methods, but all of them generate measures of predictability. Predictability is often measured in bits of surprisal, which is the negative log probability of a word (1 bit of surprisal means a word is expected to occur half the time, 2 bits is 1/4 of the time etc).

The functional form of the relationship between RTs from eye-tracking and SPR studies and the predictability of the words is linear in terms of surprisal (Goodkind & Bicknell, 2018; Luke & Christianson, 2016; Smith & Levy, 2013; Wilcox, Gauthier, Hu, Qian, & Levy, 2020). Predictability reliable correlates with reading time in the range of surprisals found in natural-sounding texts, not just for words that are extremely expected or unexpected (Smith & Levy, 2013). If Maze RTs reflect the same processing as other methods, we expect to find a similar linear relationship with surprisal.

Current experiment

The Maze task has thus far primarily been used on constructed sentences focusing on targeted effects and not on the long naturalistic passages used to assess the relationship between RT and surprisal. We tested how A-maze performs on longer naturalistic corpora.

The Natural Stories corpus (Futrell et al., 2020) consists of 10 passages of roughly 1000 words each which are designed to read fluently to native speakers. At the same time, the passages contain copious punctuation, quoted speech, proper nouns, and low frequency grammatical constructions. The corpus is accompanied by binary-choice comprehension questions, 6 per story, which we used to assess comprehension.

We tweaked the A-maze task to accommodate these longer passages and then had participants read the passages in the Maze. We compared our A-maze results with SPR data collected on the Natural Stories corpus by Futrell et al. (2020). The A-maze data and SPR comparison data allow us to address the following questions:

1. Do participants engage successfully with the A-maze task?
2. Is A-maze as reliable as SPR?
3. Are the effects of surprisal on RT linear for the A-maze task?
4. Does A-maze have less spillover than SPR?
5. Which language model best fits the RT data from A-maze and SPR?

Methods

We constructed A-maze distractors for the Natural Stories corpus (Futrell et al., 2020) and recruited 100 crowd-sourced participants to each read a story in the Maze paradigm. The materials, data, and analysis code are all available at REPO.

Task: Error-correction Maze

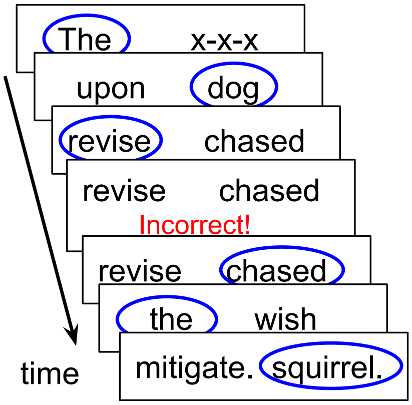

Figure 1. Schematic of error-correction Maze. A participant reads a sentence word by word, choosing the correct word at each time point (selections marked in blue ovals). When they make a mistake, an error message is displayed, so the participant can try again and continue with the sentence.

In order to support longer materials, we tweaked the Maze task, creating a new variant called error-correction Maze.

One of the benefits of the Maze task is that it forces incremental processing by having participants make an active choice about what the next word is. But what happens if they choose incorrectly? In the traditional Maze paradigm, any mistake ends the sentence, and the participant moves on to the next item (Forster et al., 2009). An advantage of this is that participants who contribute RT data are very likely to have understood the sentence up to that point. This contrasts with other methods, where determining whether participants are paying attention usually requires separate comprehension check questions, usually not used for Maze.

However, terminating sentences on errors means that we don't have RTs for words after a participant makes a mistake in an item. In traditional G-maze tasks, with hand-crafted distractors and attentive participants, errors are rare and data loss is a small issue. However, this data loss is much worse with A-maze materials and crowd-sourced participants (Boyce et al., 2020). The high errors are likely from some combination of participants guessing randomly and from auto-generated distractors that in fact fit the sentence; as Boyce et al. (2020) noted, some distractors, especially early in the sentence, were problematic and caused considerable data loss.

The high error rates could be improved by auto-generating better distractors or hand-replacing problematic ones, but that does not solve the fundamental problem with long items. Well-chosen distractors and attentive participants reduce the error rate, but the error rate will still compound over long materials. For instance, with a 1% error rate, 86% of participants would complete each 15-word sentence, but only 61% would complete a 50 word vignette, and 13% would complete a 200 word passage. In order to run longer materials, we needed something to do when participants made a mistake other than terminate the entire item.

As a solution, we introduce an *error-correction* variant of Maze shown in Figure 1. When a participant makes an error, they see an error message and must try again to select the correct option, before continuing the sentence as normal. We make error-correction Maze available as an option in a modification of the Ibex Maze implementation introduced in Boyce et al. (2020) (https://github.com/vboyce/Ibex-with-Maze). The code records both the RT to the first click and also the total RT until the correct answer is selected as separate values.

Error-correction Maze expands the types of materials that can be used with Maze to include arbitrarily long passages and cushions the impact of occasional problematic distractors. Error-correction Maze is a change in experimental procedure, and is independent of what types of distractors are used. This error-correction presentation is used here with A-maze, but could also be used with G-maze or I-maze.

Materials

We used the texts from the Natural Stories corpus (Futrell et al., 2020) and their corresponding comprehension questions. To familiarize participants with the task, we wrote a short practice passage and corresponding comprehension questions. See Appendix A for an excerpt of one of the stories.

To generate distractors, we first split the corpora up into sentences, and then ran the sentences through the A-maze generation process. We used an updated version of the codebase from Boyce et al. (2020) which had the capability to match the greater variety of punctuation present in the Natural Stories corpus (updated auto-generation code at https://github.com/vboyce/Maze). We took the auto-generated distractors as they were, without checking for quality.

Participants

We recruited 100 participants from Amazon Mechanical Turk in April 2020, and paid each participant $3.50 for roughly 20 minutes of work. We excluded data from those who did not report English as their native language, leaving 95 participants. After examining the participants performance on the task (see Results for details), we excluded data from participants with less than 80% accuracy, removing participants whose behavior was consistent with random guessing. After this exclusion, 63 participants were left.

Procedure

Participants first gave their informed consent and saw task instructions. Then they read a short practice story in the Maze paradigm and answered 2 binary-choice practice comprehension questions, before reading one main story in the A-maze task. After the story, they answered 6 comprehension questions, commented on their experience, answered optional demographic questions, were debriefed, and were given a code to enter for payment. The experiment was implemented in Ibex (https://github.com/addrummond/ibex).

Self-paced reading comparison

In addition to the texts, Futrell et al. (2020) released reading time data from a SPR study they ran in 2011. They recruited 181 participants from Amazon Mechanical Turk, most of whom read 5 of the stories. After reading each story, each participant answered 6 binary-choice comprehension questions. For comparability with A-maze, we analyze only the first story each participant read, and, in line with Futrell et al. (2020), exclude participants who got less than 5/6 of the comprehension questions correct, leaving 165.

SPR-Maze correlation

We compared the correlations between the Maze and SPR RTs to within-Maze and within-SPR correlations. For Maze, within each story, we randomly split subjects into two halves. Within each half, we calculated a per-word average RT for each word and then a per-sentence average RT across word averages. We calculated a within-Maze correlation between these two halves.

For this comparison, we downsampled the SPR data choosing a number of participants equal to the number we have for Maze to avoid differences due to dataset size. We then used the same procedure to get a within-SPR correlation. For between Maze-SPR correlation, we took the average correlation across each of the 4 pairs of Maze half and SPR half.

Modeling approach

Our analytic questions required multiple modeling approaches. To look at the functional form of the relationship between surprisal and RT data, we fit Generalized Additive Models (GAMs) to allow for non-linear relationships. GAMs are harder to interpret than linear models, so to measure effect sizes and assess spillover, we used linear mixed models. Finally, in order to determine which language model best predicts the RT data, we fit additional linear models with predictors from multiple language models to look at their relative contributions. All these models used surprisal, frequency, and length as predictors for RT. We considered these predictors from both the current and past word to account for the possibility of spillover effects in A-maze. For SPR comparisons, we included predictors from the current and past three words to account for known spillover effects. We conducted data processing and analyses using R (Version 4.0.3; R Core Team, 2020)[1].

Predictors. We created a set of predictor variables of frequency, word length, and surprisals from 4 language models. For length, we used the length in characters excluding end punctuation. For unigram frequency, we tokenized the training data from Gulordava, Bojanowski, Grave, Linzen, and Baroni (2018) and tallied up instances. We then rescaled the word counts to get the log2 frequency of occurrences per 1 billion words, so higher values indicate higher log frequencies. We got per-word surprisals for each of 4 different language models, covering a range of common architectures: a Kneser-Ney smoothed 5-gram, GRNN (Gulordava et al., 2018), Transformer-XL (Dai et al., 2019), and GPT-2 (Radford et al., n.d.), using lm-zoo (Gauthier, Hu, Wilcox, Qian, & Levy, 2020). For all of these predictors, we used both the predictor at the current word as well as lagged predictors from the previous word.

Exclusions. We excluded the first word of every sentence because it had an x-x-x distractor, leaving 9782 words. We excluded words for which we didn't have surprisal or frequency information, leaving 8489 words. We additionally excluded words that any model treated as being composed of multiple tokens (primarily words with punctuation), leaving 7512 words[2]. We excluded outlier RTs that were <100 or >5000 ms (<100 is likely a recording error, >5000 is likely the participant getting distracted). We exclude RTs from

[1] We, furthermore, used the R-packages *brms* (Version 2.14.4; Bürkner, 2017, 2018), *broom.mixed* (Version 0.2.6; Bolker & Robinson, 2020), *cowplot* (Version 1.1.1; Wilke, 2020), *gridExtra* (Version 2.3; Auguie, 2017), *here* (Version 1.0.1; Müller, 2020), *kableExtra* (Version 1.3.4; Zhu, 2021), *lme4* (Version 1.1.26; Bates et al., 2015a), *mgcv* (Version 1.8.33; Wood, 2011, 2003, 2004; Wood, N., Pya, & S"afken, 2016), *papaja* (Version 0.1.0.9997; Aust & Barth, 2020), *patchwork* (Version 1.1.1; Pedersen, 2020), *tidybayes* (Version 2.3.1; Kay, 2020), *tidymv* (Version 3.2.1; Coretta, 2021), and *tidyverse* (Version 1.3.1; Wickham et al., 2019).

[2] Surprisals should be additive, but summing the surprisals for multi-token words gave some unreasonable responses. For instance, in one story the word king!' has a surprisal of 64 under GRNN (context: The other birds gave out one by one and when the eagle saw this he thought, 'What is the use of flying any higher? This victory is in the bag and I am king!'). While GPT-2 using byte-pair encoding that can split up words into multiple parts, excluding words it split up only excluded 30 words that were not already excluded by other models.

words where mistakes occurred or which occurred after a mistake in the same sentence. We only analyzed words where we had values for all predictors, which meant that if the previous word was unknown to a model, the word was excluded because of missing values for a lagged predictor.

Model specification. For generalized additive models, we centered but did not rescale the length and frequency predictors, but left surprisal uncentered for interpretability. We used smooths for the surprisal terms and tensor effects for the frequency by length effects and interactions. To account for by-subject variation, we did bootstrapping.

For linear models, we centered all predictors. We modeled the main effects of surprisal, length, and frequency as well as surprisal x length and frequency x length interactions. For the A-maze data, we used maximal mixed effects, including by-subject slopes and a per-word-token random intercept (Barr, Levy, Scheepers, & Tily, 2013). We used weak priors (normal(1000,1000) for intercept, normal(0,500) for beta and sd, and lkj(1) for correlations) and ran models with brm (Bürkner, 2018).

For linear models of the SPR data, we were unable to fit the full mixed effects structure. The best model we could fit had by-subject random intercept, uncorrelated by-subject random slopes for surprisal, length and frequency, and a per-word-token random intercept, fit with lme4 (Bates et al., 2015b), as this structure did not fit reliably in brm.

For model comparison, we took by-item averaged data to aid in fast model fitting. We included frequency, length, and their interaction in all models. Then we fit models with either 1 or 2 sources of surprisal using lm (Bates et al., 2015b) and assessed the effect of adding the second surprisal source with an anova.

Results

Do participants engage successfully?

Our first question was whether participants could engage successfully with the error-correction Maze task. We assessed engagement by looking at participants' accuracy on the Maze task and performance on the comprehension questions.

Accuracy, or how often a participant chose the correct word over the distractor, reflects both the quality of the distractors and the focus and skill of the participant. We calculated the per-word accuracy rate for each participant and compared it against their average reaction time.[3] As seen in Figure 2A, one cluster of participants (marked in green) made relatively few errors, with some reaching 99% accuracy. This high performance confirms that the distractors were generally appropriate and shows that some participants maintained focus on the task for the whole story. These careful participants took around 1 second for each word selection, which is much slower than in eye-tracking or SPR.

[3] To avoid biasing the average if a participant took a pause before returning to the task, RTs greater than 5 seconds were excluded. This exclusion removed 260 words, or 0.27% of trials.

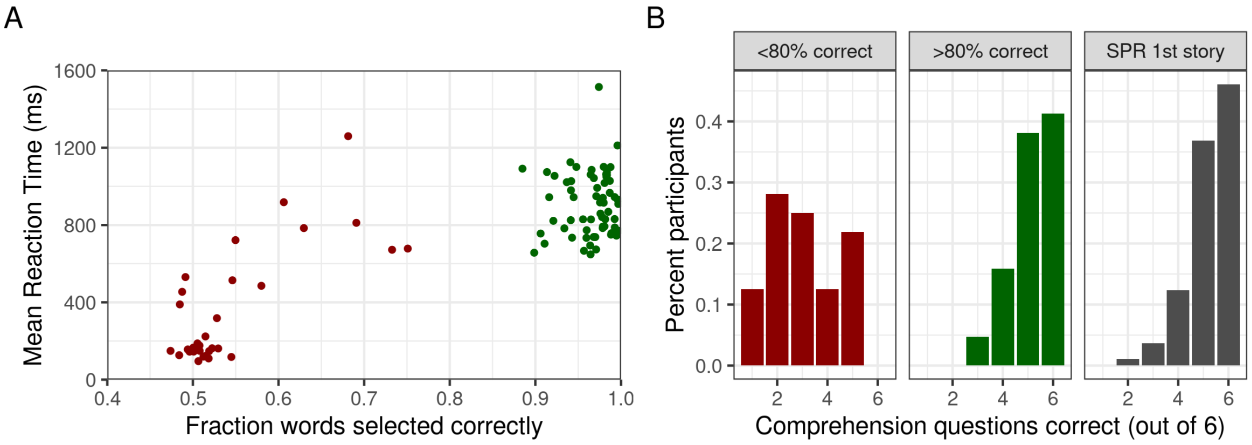

Figure 2. A. Participant's accuracy on the Maze task (fraction of words selected correctly) versus their average reaction time (in ms). Many participants (marked in green) chose the correct word >80% of the time; others (in red) appear to be randomly guessing. B. Performance on the comprehension questions. Participants with low accuracy performed poorly on comprehension questions; Participants with >80% task accuracy tended to do well; their performance was roughly comparable to the performance of SPR participants from Futrell et al. (2020) on their first stories.

Another cluster of participants (in red) sped through the task, seemingly clicking randomly. This bimodal distribution is likely due to the mix of workers on Mechanical Turk, as we did not use qualification cutoffs. We believe the high level of random guessing is an artifact of the subject population [Hauser, Paolacci, and Chandler (2018), and we expect that following current recommendations for participant recruitment, such as using qualification cutoffs or another recruitment site would result in fewer participants answering randomly (Eyal, David, Andrew, Zak, & Ekaterina, 2021; Peer, Brandimarte, Samat, & Acquisti, 2017).

To determine comprehension accuracy, we counted how many of the binary-choice comprehension questions each participant got right (out of 6). As seen in Figure 2B, most participants who were accurate on the task also did well on comprehension questions, while participants who were at chance on the task were also at chance on the comprehension questions. Participants usually answered quickly (within 10 seconds), so we do not believe they were looking up the answers on the Internet. We can't rule out that some participants may have been able to guess the answers without understanding the story. Nonetheless, the accurate answers provide preliminary evidence that people can understand and remember details of stories they read during the Maze task.

The comprehension question performance of accurate Maze participants is broadly similar to the performance of SPR participants from Futrell et al. (2020) on the first story read. Overall, 60% of Maze participants got 5 or 6 questions right (22% of low-accuracy participants and 79% of high-accuracy participants) compared to 91% of all SPR reads and 83% of 1st SPR reads. These differences cannot be directly attributed to methods, as the participant populations differed. While both studies were conducted on Mturk, the quality of Mturk data has decreased from 2011 when the SPR was collected to 2020 when the A-maze was collected (Chmielewski & Kucker, 2020).

For the remainder of the analyses, we use task performance as our exclusion metric for A-maze because it is more fine-grained and only analyze data from participants with at least 80% accuracy (in the gap between high-performers and low-performers). For the SPR comparison, we follow Futrell et al. (2020)'s criteria and exclude participants who got less than 5 of the comprehension questions correct.

Is A-maze reliable?

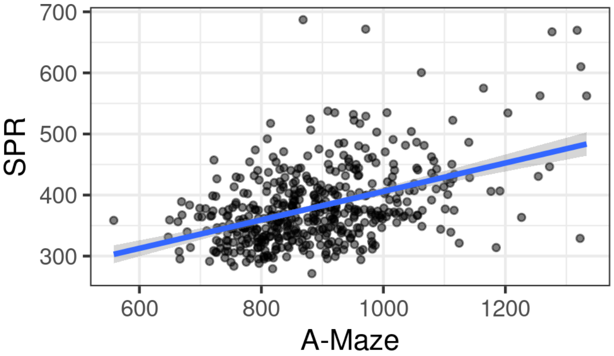

Figure 3. Correlation between SPR and Maze data. RTs were averaged across participants per word and then averaged together within each sentence. RTs in ms.

Our second question was whether A-maze is reliable. To assess reliability, we conducted split-half comparisons looking at the correlations between and within SPR and A-maze. If the methods picked up on the same effects, we would expect them to be correlated, with sentences that took longer to read in one method also taking longer in the other. We calculated the average RT at the sentence level to reduce variability from spillover patterns. The correlation between Maze and SPR was 0.25, compared to 0.23 within SPR and 0.36 within Maze. See Figure 3 for a visual comparison of overall Maze versus SPR RTs. SPR data is about as correlated with Maze as with another sample of SPR data which provides some evidence that Maze and SPR are measuring the same effects.

Are the effects of surprisal linear?

We next considered the relationship between surprisal and Maze RT. Surprisal, a measure of overall word predictability in context, is linearly related to RT in eye-tracking and SPR (Goodkind & Bicknell, 2018; Luke & Christianson, 2016; Smith & Levy, 2013; Wilcox et al., 2020). If Maze is measuring the same language processes, we would expect to see a linear relationship between surprisal and Maze RT.

To assess the shape of the RT-surprisal relationship, we fit generalized additive models (GAMs). For these models, we only included data that occurred before any mistakes in the sentence; due to limits of model vocabulary, words with punctuation and some uncommon or proper nouns were excluded. We used surprisals generated by 4 different language models for robustness. (See Methods for details on language models, exclusions, and model fit.)

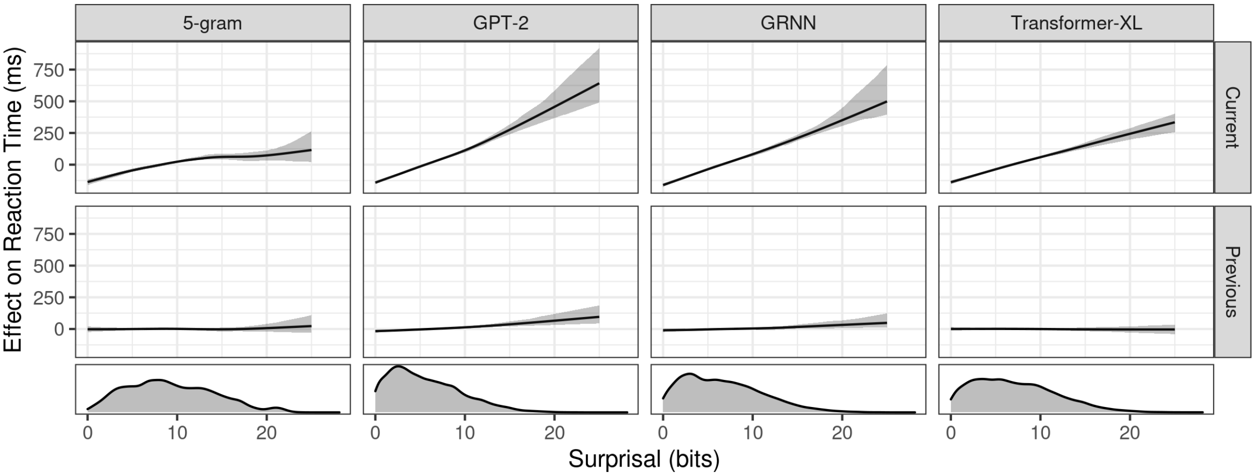

Figure 4. GAM results for the effect of current word surprisal (top) or previous word surprisal (bottom) on Maze reaction time (RT). Density of data is shown along the x-axis. For each of the 4 language models used, there is a linear relationship between current word surprisal and RT. The relationship between previous word surprisal and RT is much flatter.

The main effects of current and previous word surprisals on RT are shown in Figure 4. Note that for each of the models, high-surprisal words are rare, with much of the data from words with between 0 and 15 bits of surprisal. All 4 models show a roughly linear relationship between current word surprisal and RT, especially in the region with more data.

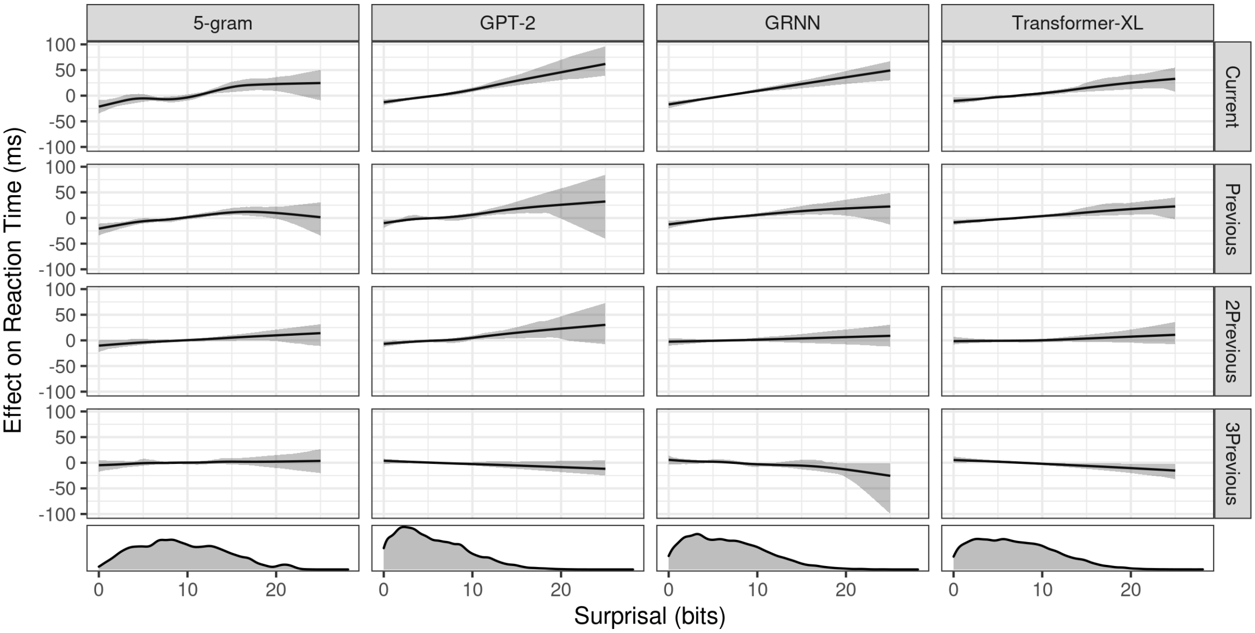

Figure 5. GAM results for the effect of current word surprisal (top) or the surprisal of an earlier word, up to 3 words back on SPR RT data (Futrell et al., 2020). Density of data is shown along the x-axis.

As a comparison, we also ran GAMs on the SPR data collected by Futrell et al. (2020).

Previous work such as Smith and Levy (2013) has found positive relationships between RT and the surprisal of earlier words for SPR, so we include predictors from the current and the 3 prior words. The relationship between surprisals and RT is shown in Figure 5; note that the y-axis range is much narrower than for Maze. Both current and previous word surprisals have a roughly linear positive relationship to RT. The surprisal of the word two back also has an influence in some models.

Comparing Maze and SPR, we see that both show a linear relationship, but Maze has much larger effects of surprisal on the current word.

Does A-maze have less spillover?

One of the main claimed advantages of the Maze task is that it has better localization and less spillover than SPR. We examined how much spillover A-maze and SPR each had by fitting linear models with predictors from current and previous words. Large effects from previous words are evidence for spillover; effects of the current word that dwarf any lagged effects is evidence for localization.

We modeled reading time as a function of surprisal, frequency, and length as well as surprisal x length and frequency x length interactions. For all of these, we included the predictors for the current and previous word, and we centered, but did not rescale, all predictors. (See Methods for more details on these predictors and model fit process.) As with the GAM models, we used surprisal calculations from 4 different language models for robustness.

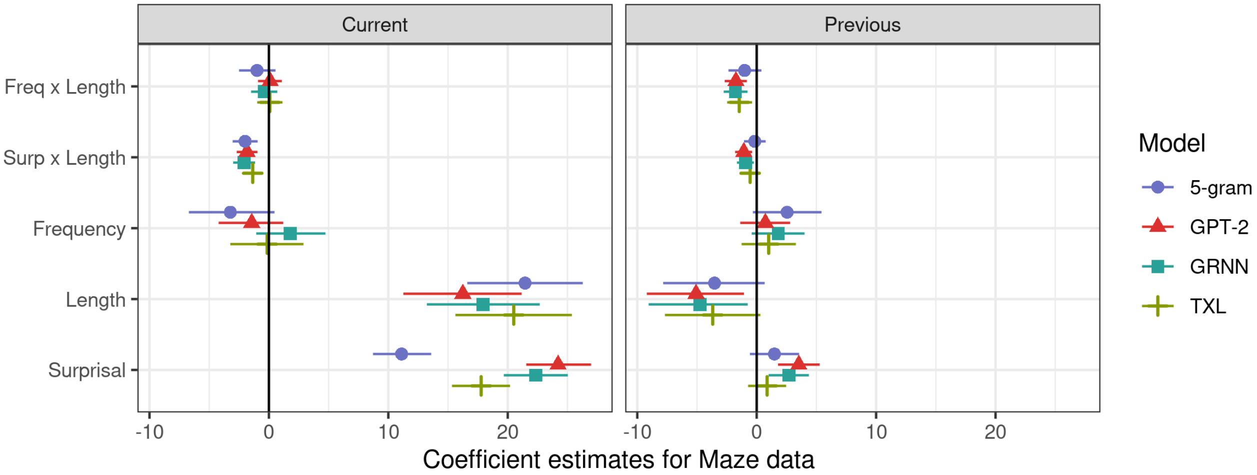

Figure 6. Point estimates and 95% credible intervals for coefficients predicted by fitted Bayesian regression models predicting A-maze RT. Units are in ms. Surprisal is per bit, length per character, and frequency per log_2 occurrence per billion words.

The Maze linear model effects are shown in Figure 6 (See also Appendix B for a table of effects). Across all models, there were consistent large effects of length and surprisal at the current word, but minimal effects of frequency. The lack of frequency effects is unexpected, but consistent with Shain (2019). There was a small interaction between surprisal and length at the current word.

Crucially, the effects of previous word predictors are close to zero, and much smaller than the effects of surprisal and length of the current word, an indication that spillover is limited and effects are strongly localized.

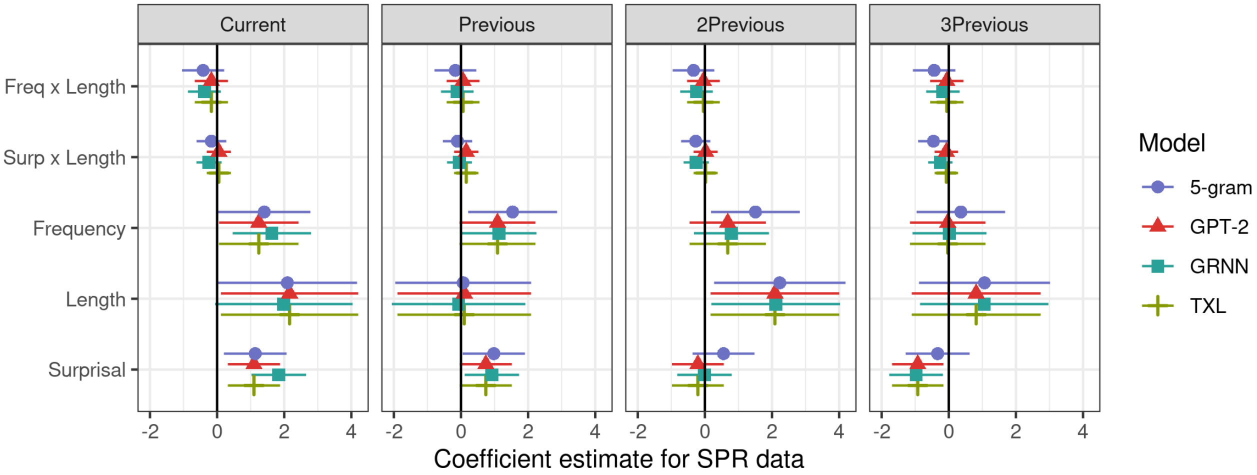

Figure 7. Point estimates and 95% credible intervals for coefficients predicted by fitted Bayesian regression models predicting SPR RT. Units are in ms. Surprisal is per bit, length per character, and frequency per log_2 occurrence per billion words.

We ran similar models for SPR, although to account for known spillover effects, we consider predictors from the current and 3 previous words. Due to issues fitting models, the details of the models differed (see Methods). The SPR coefficients are shown in Figure 7 (see also Appendix B for a table of coefficients). Surprisal, length, and frequency effects are all evident for the current word and surprisal and frequency show effects from the previous word as well. Unlike for Maze, with SPR there is not a clear diminishing of the size of the effects as one goes from current word to prior word predictors.

Whereas Maze showed surprisal effects in the 10 to 25 ms/bit range and length effects 15 to 20 ms/character range, SPR effects are about 1-2 ms per bit or character. This difference in effect size is disproportionate to the overall speed of the methods; the predicted intercept for the Maze task was roughly 880 ms and for SPR was roughly 360 ms. Thus Maze is is 2-3 times as slow as SPR but has roughly 10 times larger effects.

Which language model fits best?

Our last analysis question is whether some of the language models fit the human RT data better than others. We assessed each model's fit to A-maze data using log likelihood and R-squared. Then we did a nested model comparison, looking at whether a model with multiple surprisal predictors (ex, GRNN and GPT-2) had a better fit than a model with only one (ex GRNN alone).

As shown in Table 1, GPT-2 provides a lot of additional predictive value over each other model, GRNN provides a lot over 5-gram and TXL and a little complementary information over GPT-2. TXL provides a lot over 5-gram, and 5-gram provides little over any model. The single-model measures of log likelihood confirm this hierarchy, as GPT-2 is better than GRNN is better than TXL is better than 5-gram.

Table 1

Results of model comparisons on Maze data. Each row shows the additional predictive value gained from adding that model to another model. F values and p values from ANOVA tests between 1-surprisal-source and 2-source models are reported. We also report log likelihoods of models with only one surprisal source and the r-squared correlation between the model's predictions and the data.

Model	over 5-gram	over GRNN	over TXL	over GPT-2	Log Lik	r_squared
5-gram		2 (p=0.153)	3 (p=0.035)	0 (p=0.611)	-43817	0.16
GRNN	287 (p<0.001)		113 (p<0.001)	13 (p<0.001)	-43544	0.23
TXL	174 (p<0.001)	5 (p=0.006)		2 (p=0.137)	-43650	0.2
GPT-2	394 (p<0.001)	113 (p<0.001)	213 (p<0.001)		-43445	0.25

We followed the same process for the SPR data with results shown in Table 2. For SPR, GPT-2 and 5-gram models contain some value over each other model, which is less clear for TXL and GRNN. In terms of log likelihoods, we find that GPT-2 is better than 5-gram is better than GRNN is better than TXL, although differences are small. The relatively good fit of 5-gram models to SPR data compared with neural models matches results from Hu, Gauthier, Qian, Wilcox, and Levy (2020) and Wilcox et al. (2020), and contrasts with the Maze results, where the 5-gram model had the worst fit and did not provide additional predictive value over the other models.

As an overall measure of fit to data, we calculate multiple R-squared for the single surprisal source models for both A-maze and SPR. The models predict A-maze better than SPR with R-squared values for A-maze ranging from 0.16 for the 5-gram model to 0.25 for GPT-2. For SPR, the R-squared values range from from 0.007 to 0.011. This pattern suggests that the effect size differences are not due merely to the larger overall reading time for A-maze, but that instead A-maze is more sensitive to surprisal and length effects.

Discussion

We introduced error-correction Maze, a tweak on the presentation of Maze materials that makes Maze feasible for multi-sentence passages. We then used A-maze distractors and the error-correction Maze presentation to gather data on participants reading stories from the Natural Stories corpus in the Maze. As laid out in the Introduction, this current study addressed five main questions.

First, we found that participants could read and comprehend the 1000 word stories, despite the slowness and added overhead of reading in the Maze task. This result expands

Table 2

Results of model comparisons on SPR data. Each row shows the additional predictive value gained from adding that model to another model. F values and p values from ANOVA tests between 1-surprisal-source and 2-source models are reported. We also report log likelihoods of models with only one surprisal source and the r-squared correlation between the model's predictions and the data.

Model	over 5-gram	over GRNN	over TXL	over GPT-2	Log Lik	r_squared
5-gram		3 (p=0.032)	4 (p=0.001)	3 (p=0.033)	-51798	0.007
GRNN	7 (p<0.001)		6 (p<0.001)	2 (p=0.153)	-51790	0.009
TXL	3 (p=0.010)	0 (p=0.910)		1 (p=0.462)	-51801	0.007
GPT-2	10 (p<0.001)	5 (p<0.001)	10 (p<0.001)		-51783	0.011

the domain of materials usable with Maze beyond targeted single-sentence items to longer, naturalistic texts with sentence-to-sentence coherency.

Second, we took advantage of the pre-existing SPR corpus on Natural Stories to compare the RT profiles between Maze and SPR. Maze and SPR pick up on similar features in words, as shown by the high correlations between Maze and SPR RTs on the sentence level.

Third, we addressed whether the A-maze RT for a word showed a linear relationship with that word's surprisal. We found that A-maze RTs are linearly related to surprisal, matching the functional profile found with other incremental processing methods.

Fourth, we compared the spillover profiles between Maze and SPR. For Maze, we found large effects of the current words surprisal and length, which dwarfed any spillover effects from previous word predictors. In contrast, for SPR, we found effects of roughly equal sizes from the current and previous words. Overall, Maze is a slower task than SPR, but it also has much larger effects of length and surprisal, perhaps due to requiring more focus and thus generating less noisy data.

Lastly, we examined how different language models fare at predicting human RT data. We found that overall, the models were more predictive of the A-maze data than SPR data; however, the hierarchy of the model's predictive performance also differed between the A-maze and SPR datasets. This difference suggests that how predictive a language model is of RTs may depend on the task. Further comparisons between different processing methods on the same materials could be useful for identifying how task demands influence language processing (ex. Bartek et al., 2011).

Overall, A-maze has good localization, although some models showed small but statistically reliable effects of the past word. On the whole, however, our results support the idea that Maze forces language processing to be close to word-by-word, and thus the Maze task can be used under the assumption that the RT of a word primarily reflects its own properties and not those of earlier words.

Limitations

While we expect these patterns of results reflect features of the A-maze task, the effects could be moderated by quirks of the materials or the participant population. We excluded a large number of participants for having low accuracy on the task and appearing to guess randomly. We compared RTs collected on the A-maze task to SPR RTs previously collected on the same corpus, but we did not randomly assign participants to SPR and Maze conditions. This study suggests that A-maze is a localized and widely-usable method, but only broader applications can confirm these findings.

Future directions

Error-correction Maze reduces perverse incentives from the desire to complete the task quickly compared to traditional Maze, but clicking randomly is still more efficient than doing the task. In discussing this work, we received the suggestion that one way to further disincentivize random clicking would be to add a pause when a participant makes a mistake, forcing them to wait some short period of time (ex 500ms) before correcting their mistake. This delay would make randomly hitting buttons slower than doing the task as intended.

Error-correction Maze records RTs for words after a participant makes a mistake in the sentence. In our analyses, we excluded this post-error data, but we believe it is an open question whether data from after a participant makes a mistake is usable. That is, does it show the same profile as RTs from pre-error words, or are there traces from recovering from the mistake, and if so, how long do these effects take to fade? Whether post-mistake data is high-quality and trustworthy enough to be included in analyses is hard-to-assess; if it can be used, it would make the Maze task more data efficient.

The Maze task is versatile and can be used or adapted for a wide range of materials and questions of interest. The extreme incrementality makes the Maze task a good target for any question that requires precisely determining the locus of incremental processing difficulty. We encourage researchers to use Maze as an incremental processing method, alone or in comparison with other methods, and we suggest that the error-correction mode be the default choice for presenting Maze materials.

References

Auguie, B. (2017). *GridExtra: Miscellaneous functions for "grid" graphics.* Retrieved from https://CRAN.R-project.org/package=gridExtra

Aust, F., & Barth, M. (2020). *papaja: Create APA manuscripts with R Markdown.* Retrieved from https://github.com/crsh/papaja

Barr, D. J., Levy, R., Scheepers, C., & Tily, H. J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, *68*(3), 255–278.

Bartek, B., Lewis, R. L., Vasishth, S., & Smith, M. R. (2011). In search of on-line locality effects in sentence comprehension. *Journal of Experimental Psychology: Human Perception & Performance*, *37*(5), 1178.

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015a). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*(1), 1–48. https://doi.org/10.18637/jss.v067.i01

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015b). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*(1), 1–48.

Bolker, B., & Robinson, D. (2020). *Broom.mixed: Tidying methods for mixed models.* Retrieved from https://CRAN.R-project.org/package=broom.mixed

Boyce, V., Futrell, R., & Levy, R. P. (2020). Maze Made Easy: Better and easier measurement of incremental processing difficulty. *Journal of Memory and Language*, *111*, 104082.

Bürkner, P.-C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, *80*(1), 1–28. https://doi.org/10.18637/jss.v080.i01

Bürkner, P.-C. (2018). Advanced Bayesian multilevel modeling with the R package brms. *The R Journal*, *10*(1), 395–411. https://doi.org/10.32614/RJ-2018-017

Bürkner, P.-C. (2018). Advanced bayesian multilevel modeling with the r package brms. *The R Journal*, *10*(1), 395–411.

Chmielewski, M., & Kucker, S. C. (2020). An MTurk Crisis? Shifts in Data Quality and the Impact on Study Results. *Social Psychological and Personality Science*, *11*(4), 464–473. https://doi.org/10.1177/1948550619875149

Coretta, S. (2021). *Tidymv: Tidy model visualisation for generalised additive models.* Retrieved from https://CRAN.R-project.org/package=tidymv

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019).

Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context. *arXiv:1901.02860 [Cs, Stat]*. Retrieved from http://arxiv.org/abs/1901.02860

Eyal, P., David, R., Andrew, G., Zak, E., & Ekaterina, D. (2021). Data quality of platforms and panels for online behavioral research. *Behav Res*. https://doi.org/10.3758/s13428-021-01694-3

Forster, K. I., Guerrera, C., & Elliot, L. (2009). The maze task: Measuring forced incremental sentence processing time. *Behavior Research Methods*, *41*(1), 163–171.

Frazier, L., & Rayner, K. (1982). Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. *Cognitive Psychology*, *14*, 178–210.

Freedman, S. E., & Forster, K. I. (1985). The psychological status of overgenerated sentences. *Cognition*, *19*(2), 101–131.

Futrell, R., Gibson, E., Tily, H. J., Blank, I., Vishnevetsky, A., Piantadosi, S. T., & Fedorenko, E. (2020). The Natural Stories corpus: A reading-time corpus of English texts containing rare syntactic constructions. *Lang Resources & Evaluation*.

Gauthier, J., Hu, J., Wilcox, E., Qian, P., & Levy, R. (2020). SyntaxGym: An Online Platform for Targeted Evaluation of Language Models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations* (pp. 70–76). Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-demos.10

Goodkind, A., & Bicknell, K. (2018). Predictive power of word surprisal for reading times is a linear function of language model quality. In *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)* (pp. 10–18). Salt Lake City, Utah: Association for Computational Linguistics. https://doi.org/10.18653/v1/W18-0102

Grodner, D., & Gibson, E. (2005). Some consequences of the serial nature of linguistic input. *Cognitive Science*, *29*(2), 261–290.

Gulordava, K., Bojanowski, P., Grave, E., Linzen, T., & Baroni, M. (2018). Colorless green recurrent networks dream hierarchically. In *Proceedings of the 2018 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 1195–1205).

Hauser, D., Paolacci, G., & Chandler, J. J. (2018). *Common Concerns with MTurk as a Participant Pool: Evidence and Solutions* (Preprint). PsyArXiv. https://doi.org/10.31234/osf.io/uq45c

Hu, J., Gauthier, J., Qian, P., Wilcox, E., & Levy, R. P. (2020). A Systematic Assessment of

Syntactic Generalization in Neural Language Models. *arXiv:2005.03692 [Cs]*. Retrieved from http://arxiv.org/abs/2005.03692

Kay, M. (2020). *tidybayes: Tidy data and geoms for Bayesian models.* https://doi.org/10.5281/zenodo.1308151

Kliegl, R., Grabner, E., Rolfs, M., & Engbert, R. (2004). Length, frequency, and predictability effects of words on eye movements in reading. *European Journal of Cognitive Psychology*, *16*(1-2), 262–284.

Koornneef, A. W., & van Berkum, J. J. A. (2006). On the use of verb-based implicit causality in sentence comprehension : Evidence from self-paced reading and eye tracking. *Journal of Memory and Language*, *54*(4), 445–465.

Levy, R., Bicknell, K., Slattery, T., & Rayner, K. (2009). Eye movement evidence that readers maintain and act on uncertainty about past linguistic input. *PNAS*, *106*(50), 21086–21090.

Luke, S. G., & Christianson, K. (2016). Limits on lexical prediction during reading. *Cognitive Psychology*, *88*, 22–60. https://doi.org/10.1016/j.cogpsych.2016.06.002

MacDonald, M. C. (1993). The interaction of lexical and syntactic ambiguity. *Journal of Memory and Language*, *32*, 692–715.

Müller, K. (2020). *Here: A simpler way to find your files.* Retrieved from https://CRAN.R-project.org/package=here

Pedersen, T. L. (2020). *Patchwork: The composer of plots.* Retrieved from https://CRAN.R-project.org/package=patchwork

Peer, E., Brandimarte, L., Samat, S., & Acquisti, A. (2017). Beyond the Turk: Alternative platforms for crowdsourcing behavioral research. *Journal of Experimental Social Psychology*, *70*, 153–163. https://doi.org/10.1016/j.jesp.2017.01.006

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (n.d.). Language Models are Unsupervised Multitask Learners, 24.

Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, *124*(3), 372–422.

Rayner, K., Ashby, J., Pollatsek, A., & Reichle, E. D. (2004). The Effects of Frequency and Predictability on Eye Fixations in Reading: Implications for the E-Z Reader Model. *Journal of Experimental Psychology: Human Perception and Performance*, *30*(4), 720–732.

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/

Shain, C. (2019). A large-scale study of the effects of word frequency and predictability in naturalistic reading. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4086–4094). Minneapolis, Minnesota: Association for Computational Linguistics.

Shain, C., & Schuler, W. (2018). Deconvolutional Time Series Regression: A Technique for Modeling Temporally Diffuse Effects. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (pp. 2679–2689). Brussels, Belgium: Association for Computational Linguistics. https://doi.org/10.18653/v1/D18-1288

Sloggett, S., Handel, N. V., & Rysling, A. (2020). A-maze by any other name. In *CUNY*.

Smith, N. J., & Levy, R. (2013). The effect of word predictability on reading time is logarithmic. *Cognition*, *128*(3), 302–319.

Staub, A. (2010). Eye movements and processing difficulty in object relative clauses. *Cognition*, *116*, 71–86.

Traxler, M. J., Morris, R. K., & Seely, R. E. (2002). Processing subject and object relative clauses: Evidence from eye movements. *Journal of Memory and Language*, *47*, 69–90.

Vani, P., Wilcox, E. G., & Levy, R. (2021). Using the Interpolated Maze Task to Assess Incremental Processing in English Relative Clauses. *Proceedings of the Annual Meeting of the Cognitive Science Society*, *43*(43).

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., … Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Wilcox, E. G., Gauthier, J., Hu, J., Qian, P., & Levy, R. (2020). On the Predictive Power of Neural Language Models for Human Real-Time Comprehension Behavior. *arXiv:2006.01912 [Cs]*. Retrieved from http://arxiv.org/abs/2006.01912

Wilcox, E., Vani, P., & Levy, R. (2021). A Targeted Assessment of Incremental Processing in Neural Language Models and Humans. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 939–952). Online: Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.76

Wilke, C. O. (2020). *Cowplot: Streamlined plot theme and plot annotations for 'ggplot2'*. Retrieved from https://CRAN.R-project.org/package=cowplot

Witzel, N., Witzel, J., & Forster, K. (2012). Comparisons of online reading paradigms: Eye tracking, moving-window, and maze. *Journal of Psycholinguistic Research*, *41*(2), 105–128.

Wood, S. N. (2003). Thin-plate regression splines. *Journal of the Royal Statistical Society (B)*, *65*(1), 95–114.

Wood, S. N. (2004). Stable and efficient multiple smoothing parameter estimation for generalized additive models. *Journal of the American Statistical Association*, *99*(467), 673–686.

Wood, S. N. (2011). Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society (B)*, *73*(1), 3–36.

Wood, S. N., N., Pya, & S"afken, B. (2016). Smoothing parameter and model selection for general smooth models (with discussion). *Journal of the American Statistical Association*, *111*, 1548–1575.

Zhu, H. (2021). *KableExtra: Construct complex table with 'kable' and pipe syntax.* Retrieved from https://CRAN.R-project.org/package=kableExtra

Appendix A

The beginning of one of the stories. This excerpt is the first 200 words of a 1000 word story.

Tulip mania was a period in the Dutch Golden Age during which contract prices for bulbs of the recently introduced tulip reached extraordinarily high levels and then suddenly collapsed. At the peak of tulip mania in February sixteen thirty-seven, tulip contracts sold for more than ten times the annual income of a skilled craftsman. It is generally considered the first recorded economic bubble. The tulip, introduced to Europe in the mid sixteenth century from the Ottoman Empire, became very popular in the United Provinces, which we now know as the Netherlands. Tulip cultivation in the United Provinces is generally thought to have started in earnest around fifteen ninety-three, after the Flemish botanist Charles de l'Ecluse had taken up a post at the University of Leiden and established a botanical garden, which is famous as one of the oldest in the world. There, he planted his collection of tulip bulbs that the Emperor's ambassador sent to him from Turkey, which were able to tolerate the harsher conditions of the northern climate. It was shortly thereafter that the tulips began to grow in popularity. The flower rapidly became a coveted luxury item and a status symbol, and a profusion of varieties followed.

The first 2 out of the 6 comprehension questions.

When did tulip mania reach its peak? 1630's, 1730's

From which country did tulips come to Europe? Turkey, Egypt

Appendix B

Table 3

Predictions from fitted Bayesian regression models. All terms were centered, but not rescaled. Units are in ms. Surprisal is per bit, length per character, and frequency per log$_2$ occurance per billion words. Interval is 2.5th quantile to 97.5th quantile of model draws.

Term	5-gram	TXL	GRNN	GPT-2
Intercept	876 [840.4, 910.9]	880 [842.8, 914.9]	876.8 [840.1, 911.5]	878.5 [845.6, 911.6]
Surprisal	11.1 [8.7, 13.6]	17.8 [15.3, 20.2]	22.3 [19.7, 25]	24.2 [21.5, 27]
Length	21.4 [16.6, 26.3]	20.5 [15.6, 25.4]	17.9 [13.2, 22.7]	16.2 [11.3, 21.2]
Frequency	-3.2 [-6.7, 0.5]	-0.1 [-3.2, 2.9]	1.8 [-1.1, 4.7]	-1.4 [-4.2, 1.2]
Surp x Length	-2 [-3, -0.9]	-1.4 [-2.1, -0.6]	-2.1 [-3, -1.2]	-1.8 [-2.7, -1]
Freq x Length	-1 [-2.5, 0.6]	0.1 [-1, 1.1]	-0.4 [-1.5, 0.7]	0.1 [-0.9, 1.1]
Past Surprisal	1.5 [-0.6, 3.5]	0.9 [-0.7, 2.5]	2.7 [1, 4.4]	3.5 [1.8, 5.3]
Past Length	-3.5 [-7.8, 0.7]	-3.7 [-7.7, 0.3]	-4.8 [-9, -0.8]	-5.1 [-9.2, -1.1]
Past Freq	2.5 [-0.3, 5.4]	1 [-1.3, 3.3]	1.8 [-0.4, 4]	0.7 [-1.4, 2.8]
Past Surp x Length	-0.2 [-1.1, 0.8]	-0.5 [-1.2, 0.2]	-0.9 [-1.7, -0.2]	-1.1 [-1.8, -0.4]
Past Freq x Length	-1 [-2.4, 0.4]	-1.5 [-2.5, -0.4]	-1.8 [-2.8, -0.8]	-1.7 [-2.7, -0.8]

Table 4

Predictions from fitted regression models for SPR data. All terms were centered, but not rescaled. Units are in ms. Surprisal is per bit, length per character, and frequency per log$_2$ occurance per billion words. Uncertainty interval is +/- 1.97 standard error.

Term	5-gram	TXL	GRNN	GPT-2
Intercept	361.6 [344.5, 378.6]	363.9 [346.9, 380.9]	363.8 [346.8, 380.8]	363.9 [346.9, 380.9]
Surprisal	1.1 [0.2, 2.1]	1.1 [0.3, 1.9]	1.8 [1, 2.7]	1.1 [0.3, 1.9]
Length	2.1 [0, 4.2]	2.2 [0.1, 4.2]	2 [-0.1, 4]	2.2 [0.1, 4.2]
Frequency	1.4 [0, 2.8]	1.2 [0.1, 2.4]	1.6 [0.5, 2.8]	1.2 [0.1, 2.4]
Surp x Length	-0.2 [-0.6, 0.3]	0.1 [-0.3, 0.4]	-0.2 [-0.6, 0.1]	0.1 [-0.3, 0.4]
Freq x Length	-0.4 [-1, 0.2]	-0.2 [-0.7, 0.3]	-0.4 [-0.9, 0.1]	-0.2 [-0.7, 0.3]
Past Surprisal	1 [0.1, 1.9]	0.7 [0, 1.5]	0.9 [0.1, 1.7]	0.7 [0, 1.5]
Past Length	0.1 [-2, 2.1]	0.1 [-1.9, 2.1]	-0.1 [-2.1, 1.9]	0.1 [-1.9, 2.1]
Past Freq	1.5 [0.2, 2.9]	1.1 [0, 2.2]	1.1 [0, 2.2]	1.1 [0, 2.2]
Past Surp x Length	-0.1 [-0.5, 0.3]	0.2 [-0.2, 0.5]	0 [-0.4, 0.3]	0.2 [-0.2, 0.5]
Past Freq x Length	-0.2 [-0.8, 0.5]	0.1 [-0.4, 0.6]	-0.1 [-0.6, 0.4]	0.1 [-0.4, 0.6]
2Past Surprisal	0.6 [-0.4, 1.5]	-0.2 [-1, 0.6]	0 [-0.8, 0.8]	-0.2 [-1, 0.6]
2Past Length	2.2 [0.3, 4.2]	2.1 [0.2, 4]	2.1 [0.2, 4]	2.1 [0.2, 4]
2Past Freq	1.5 [0.2, 2.8]	0.7 [-0.5, 1.8]	0.8 [-0.3, 1.9]	0.7 [-0.5, 1.8]
2Past Surp x Length	-0.3 [-0.7, 0.2]	0 [-0.3, 0.4]	-0.3 [-0.6, 0.1]	0 [-0.3, 0.4]
2Past Freq x Length	-0.3 [-1, 0.3]	0 [-0.5, 0.4]	-0.3 [-0.7, 0.2]	0 [-0.5, 0.4]
3Past Surprisal	-0.3 [-1.3, 0.6]	-0.9 [-1.7, -0.2]	-1 [-1.8, -0.2]	-0.9 [-1.7, -0.2]
3Past Length	1.1 [-0.9, 3]	0.8 [-1.1, 2.7]	1.1 [-0.9, 3]	0.8 [-1.1, 2.7]
3Past Freq	0.4 [-1, 1.7]	0 [-1.2, 1.1]	0 [-1.1, 1.1]	0 [-1.2, 1.1]
3Past Surp x Length	-0.5 [-0.9, 0]	-0.1 [-0.4, 0.3]	-0.3 [-0.6, 0.1]	-0.1 [-0.4, 0.3]
3Past Freq x Length	-0.4 [-1.1, 0.2]	-0.1 [-0.6, 0.4]	-0.2 [-0.7, 0.3]	-0.1 [-0.6, 0.4]