

Pretraining Language Models for Diachronic Linguistic Change Discovery

Elisabeth Fittschen¹, Sabrina Li², Tom Lippincott², Leshem Choshen³, Craig Messner²

¹University of Hamburg, Germany

²Center for Digital Humanities, Johns Hopkins University, USA

³IBM Research, MIT, USA

Abstract

Large language models (LLMs) are increasingly used as knowledge discovery tools. Humanistic disciplines like historical linguistics and literary studies have shown interest in this capability. These fields often construct arguments on the basis of distinctions between phenomena like time-period or genre. Such methodological investments complicate reliance on LLMs pretrained over large sets of broadly-collected data. We show that efficient pretraining techniques produce useful models of semantic change over modest historical corpora without allowing potential contamination from anachronistic data. We verify that these trained-from-scratch models better respect historical divisions and are more computationally efficient compared to the standard approach of fine-tuning an existing LLM. We compare the trade-offs in general linguistic fluency versus detecting and characterizing various forms of linguistic change, and provide a pipeline implementation of our approach that can be readily adapted and applied to a wide range of diachronic phenomena.

1 Introduction

Certain fields of research invest heavily in the importance of boundaries that demarcate their objects of study into groups. These distinctions might include general phenomena like the arrow of time (e.g. diachronic change in linguistics) or specific boundaries drawn from traditional means of practice (e.g. forms of poetry in literary studies).

Such methodological investments are somewhat at odds with the dominant modern technology for language research, pretraining large language models (LLMs) (Underwood et al., 2025). LLMs are at least in part successful due to their omnivorous nature (Kaplan et al., 2020); they develop general skills by consuming as diverse and as large a corpus as possible (Polo et al., 2024). In contrast, our target fields are characterized by both limited data

and particular interests. For example, diachronic linguistics cares more about time-specific aspects of language rather than general model capability. LLMs do have some ability to divide information (for example, produce a haiku and not a limerick in a zero-shot setting) (Cheng et al., 2024; Ifergan et al., 2024). However, prompting or other elicitation techniques may not offer immediate evidence that a particular output relied only on knowledge appropriate to a period or division. We propose a straightforward solution. To ensure a model's weights contain no proscribed information, you can train a model on a restricted corpus of your own choosing.

We show that pretraining under a tight academic budget of data (and compute) proves surprisingly effective when performed using the efficient methods developed by the BabyLM community (Hu et al., 2024). Although those techniques are designed for data efficiency and cognitively-plausible pretraining, we find that BabyLlama-2 (Tastet and Timiryasov, 2024) is also a useful recipe for academic pretraining.

We leverage this form of efficiency by developing a multiple-model approach that offers exploratory access to corpus-level hypotheses concerning lexical change, non-lexical (grammatical and morphological) change, and word sense introduction/obsolescence. Specifically, we train a series of 5 models each with a pretraining dataset of 10 million tokens drawn from consecutive historical periods. We evaluate these pretrained models (*scratch*), as well as larger models finetuned on the same data slices (*finetuned*), using standard metrics and evaluation sets. We also evaluate both model sets along the axis of historical specificity by determining how much information "leakage" between periods each approach allows. Finally, we demonstrate our scratch method's ability to analyze diachronic linguistic change using a novel cloze evaluation set, qualitative analysis and use-

case examples. We find that:

1. The scratch models train nearly two times faster than DoRA finetuned models while retaining adequate performance for many tasks
2. The finetuned models "leak" information across time periods in a way that our models do not, justifying a tradeoff of test set performance for specificity
3. The full battery of scratch models can be used to generate hypotheses about linguistic change across our corpus

These qualities render our approach useful for hypothesis discovery and exploration in fields that demand firm lines of epistemological demarcation.

2 Related work

2.1 LLM analysis of lexical semantic change

Multiple works suggest that LLMs could serve linguistic studies in flexible ways not imagined before (Warstadt and Bowman, 2022), such as simulating human subject responses (Wilcox et al., 2020; Trott, 2024; Aher et al., 2022) or modeling language acquisition (Hu et al., 2024; Warstadt et al., 2023). We draw upon this thinking, calling for pretraining as a now feasible and promising way of contrasting corpora and aiding linguistic queries, broadly, and specifically for this in diachronic lexical semantic change. Lexical semantic change is a field dedicated to finding words that changed in meaning. Work in this field have employed LLMs, initially embeddings from masked language models (MLMs) (Periti and Montanelli, 2024). This approach leads to issues processing and aligning embeddings (Schlechtweg et al., 2020) and potentially complicates change discovery outside of benchmark environments (Umarova et al., 2025). Causal modeling has also been employed for diachronic change detection. However, prior approaches largely use causal models to modify (Periti et al., 2024b) or augment (Periti et al., 2024a) text samples before processing them with downstream MLMs. Other work has used inference from pretrained autoregressive models to directly measure change, relying on in-context learning and RAG to guide the model’s judgments (Periti et al., 2024c; Hagen, 2025). While these approaches have shown potential, our approach of pretraining contrasting efficient models both guarantees division between domains and offers the possibility that the

models capture features that extend past lexical meaning.

2.2 Efficient finetuning

Increased finetuning costs have led to the development of parameter-efficient rank-adaptor finetuning techniques like LoRA (Hu et al., 2022) and DoRA (Liu et al., 2024). As these techniques are the most similar in compute and token demands to our approach we use DoRA finetuning to produce comparison performance and specificity scores. We select DoRA over LoRA due to its generally reported higher performance.

2.3 Domain-specific language modeling

Previous embedding models approaches to domain specific modeling have often chosen BERT-like architectures, employing finetuning (Hosseini et al., 2021; Qiu and Xu, 2022) and pretraining (Beck and Köllner, 2023; Manjavacas and Fonteyn, 2022). However, beyond being non-causal, these strategies employ large sets of data, limiting their ability to be adapted for smaller domains of interest. One project trains a recent-history-aware (2011 to 2022) model on GPT2, but does so in order to detect knowledge-level analogies rather than provide a methodological lever, and also employs a large dataset (Drinkall et al., 2024).

2.4 Evaluating and guaranteeing historically-specific model knowledge

Datasets for evaluating diachronic model knowledge have previously focused on historical performance, both linguistic (Manjavacas and Fonteyn, 2022) and at the level of knowledge (Dhingra et al., 2022; Piryani et al., 2024), with these latter sets typically being structured in QA form.

3 Experiments

3.1 Setup

Training data. We employ a multistage pipeline to prepare time-period specific training data. This pipeline integrates three sources to estimate the publication date of each document found in the Project Gutenberg collection. (1) Author information sourced from WikiData (*WD*) (2) Work metadata found in the Project Gutenberg Catalog (*PGC*) and (3) Inference performed by *LLMs*

We define our overall historical range as the years 1750-1940, inclusive. We use a fuzzy string matching system, described in Appendix B to align

Text	Sense Year
“They had a bunch of crazy ideas that would never work ”	1599
“I tried to call the operator but the phone was dead ”	1882
“You know how it is. I’m not into ironing. It’s not my thing ”	1936
“Let’s go where there’s some life. Whatta ya say? Hey baby, I’m down ”	1952

Table 1: Cloze task examples and the year when the word sense first appeared

authors from WD to PGC works. We acquire publication dates for each author-aligned work by prompting an instruction-tuned LLM. We evaluate this work-date attribution approach over a variety of open- and closed-source LLMs by calculating their performance against a gold-annotated test set consisting of works published from 1550-1850. While closed-source LLMs perform best, Llama3.3-70B (Grattafiori et al., 2024), quantized to 4 bits using the BitsAndBytes library (Belkada et al., 2023) performs well enough to justify its use (see Appendix B for evaluation details).

We use this information to split the full corpus into 5 diachronic slices. We set the boundaries for these slices by negotiating between minimizing their duration and obtaining 10 million training tokens for each split. We further reserve 5 million tokens from each split for testing and 1 million for validation during training. This results in 5 equal subcorpora for the time periods 1750-1820, 1820-1850, 1850-1880, 1880-1910, and 1910-1940.

3.2 Procedure

Model training. We employ two primary training approaches over each split of the historical data: (1) *finetuned* models adapted from a larger pretrained model, and (2) *scratch* models trained solely on the historical datasets. We train 3 full batteries of scratch models with unique random initializations and training set shuffles. We present all scratch results with a 1 standard deviation uncertainty region calculated from these runs.

We train the finetuned models using DoRA adapters on top of a Llama3-8B backbone. We train the scratch models using the BabyLlama-2 recipe, which employs a distillation approach. Ultimately, training the scratch BabyLlama-2 models proved quicker and more efficient than the DoRA finetuning process (see Appendix A).

As an additional point of comparison, we also produce a battery of models derived from the released checkpoint of BabyLlama-2. We DoRA fine-tune this released checkpoint, which was pretrained

over the BabyLM data mixture, with each slice of our historical data, resulting in the *small* battery of models. This small set underperformed the scratch and finetuned approaches, and was consequently set aside (see Appendix E) ¹

4 Evaluation

We evaluate the scratch and finetuned models using perplexity, a modified version of the BabyLM evaluation pipeline, and a novel cloze-structured evaluation set. Additionally, when appropriate, we contextualize the scratch evaluations using the released checkpoint of BabyLlama-2 (*unadapted* BabyLlama-2) to calibrate expectations of typical performance of data-efficient models. Similarly, when appropriate, we contextualize the finetuned evaluations using distribution Llama-3-8B as a baseline (*unadapted* Llama).

We use all three of these evaluation approaches to characterize a given model’s general **fluency** and **historical specificity**. We further employ the cloze set to demonstrate the scratch approach’s **exploratory potential**.

4.1 Fluency and Historical Specificity

Perplexity. We calculate perplexity for the scratch and finetuned models using both the test set drawn from its own historical period and each test set from every other historical slice. We use the former perplexity score to measure model fluency, and the latter scores, which we call “cross-time perplexity” as a measure of historical specificity.

We consider a fluent and well-adapted model to have absolutely low perplexity over the test set drawn from its own time period and relatively high perplexity over those drawn from all other time periods.

BabyLM Evaluation Pipeline: BLiMP. The BabyLM evaluation pipeline provided by Choshen et al. (2024) is a version of EleutherAI’s Im-

¹Models and datasets are found [here](#), the code repository is found [here](#).

Model	Sentence
1750 to 1820	with whom he talked in the station at fort wayne interested him
1820 to 1850	with whom he talked in the station at fort wayne interested him
1850 to 1880	with whom he talked in the station at fort wayne interested him
1880 to 1910	with whom he talked in the station at fort wayne interested him
1910 to 1940	with whom he talked in the station at fort wayne interested him

Table 2: Normalized perplexities for scratch models, lighter red signifies higher surprisal.

evaluation-harness (Gao et al., 2023), modified to support the evaluation of models trained over a token-limited corpus, in our case the in-vocabulary overlap for all of the time slices. We ensure all test samples are in-vocabulary by filtering to the set of samples where every word appears twice in all model training sets (*maximally filtered*). The pipeline supports evaluation over BLiMP, GLUE (Wang et al., 2018) and EWoK (Ivanova et al., 2024) of which we solely utilize BLiMP (Warstadt et al., 2020). Concretely, BLiMP tests the model’s ability to understand different linguistic phenomena, which we aggregate to measure model fluency. The maximally filtered BLiMP evaluation set retains 25373 of standard BLiMP’s 67000 examples. We report the distribution of retained subtasks in Appendix D.

Novel word sense cloze evaluation set We construct this dataset using the Oxford English Dictionary (*OED*), which catalogs English words and their respective word senses. For each word sense the OED provides the year of its first attested usage, as well as example sentences illustrating the word sense in context. To generate a usable cloze task for next-token prediction models without the ability to follow instructions, the masked words need to be located at the end of the sentences. We select sentences where the word in question appears within the last 10% of characters. For practicality, we further restrict the dataset to words the OED doesn’t consider exceptionally rare, specifically ones appearing once every thousand to a million words (Table 11 in Appendix C). We filter the dataset in the same manner as the BLiMP task, removing samples with uncommon words that have less than two occurrences in any training set.

We perform evaluation by generating the top k one-word model responses to each datapoint. We achieve this using a custom Transformers (Wolf et al., 2020) LogitsProcessor, which redistributes the probability mass of tokens initiating a new word to the EOS token. In combination with probability-

based beam search (length penalty set to zero) this method efficiently approximates the top k one word responses. We were unable to find a similar approach in the literature. Example tasks are shown in Table 1.

We use this task to evaluate historical specificity by defining a measure called ‘leakage’. Leakage measures how much knowledge a model $m_{(t_0, t_1)}$ modeling a specific time period (t_0, t_1) has of future word senses. An ideal historical model should be familiar with word senses introduced before its knowledge cutoff year, and unfamiliar with those after. Let $\text{Top}_k(m_{(t_0, t_1)}, x)$ be the set of the model’s first k completions for cloze task x , and $w^*(x)$ denote the correct answer. For each model we split the cloze tasks into T_{t_1} , word senses introduced before or in year t_1 , and F_{t_1} for senses introduced after. We consider a model to complete a cloze task if the targeted word is within the top 100 answers:

$$C_{m(t_0, t_1), k} = \{x : w^*(x) \in \text{Top}_k(m_{(t_0, t_1)}, x)\}$$

Leakage is defined as accuracy over future senses:

$$l_{m(t_0, t_1)} = \frac{|C_{m(t_0, t_1), 100} \cap F_{t_1}|}{|F_{t_1}|}$$

However, we observed that since new word senses don’t emerge arbitrarily, stronger language models appear to have higher leakage. Some future cloze tasks can be correctly completed using prior word senses, such as ‘silver’ in Table 4. To filter out this effect, we introduce ‘recall normalized leakage’ which normalizes leakage using the model’s pre-cutoff recall:

$$r_{m(t_0, t_1)} = \frac{|C_{m(t_0, t_1), 100} \cap T_{t_1}|}{|T_{t_1}|}$$

The recall normalized leakage is defined as:

$$RNL_{m(t_0, t_1)} = \frac{l_{m(t_0, t_1)}}{r_{m(t_0, t_1)}}$$

More details on sense distribution and evaluation can be found in Appendix C.

4.2 Exploratory Analysis

We perform bottom-up and top down exploratory analysis. To perform bottom-up analysis we contrast the min-max normalized log perplexity of the models over a given sentence. Despite the models having different baseline perplexities, their normalized log perplexity follows a similar trajectory, with the exception of words particularly characteristic (time period-specific) for a model’s dataset. This phenomenon is shown in Table 2, where a significant shift can be seen for "station", which lowers in perplexity as the railway system is widely adopted during the 1840s and 50s. We use this perplexity data to generate candidates for word sense change, motivated by the notion that words whose later sense has not yet emerged should prove more perplexing for earlier models. We also demonstrate how the OED-derived cloze task can be used in a top-down fashion to characterize diachronic change in words known to be historically-sensitive.

5 Results and discussion

5.1 Fluency and Historical Specificity

The unadapted and finetuned models are fluent, but lack historical specificity. The unadapted baseline models show slight time-period preferences but are logically a-historical (Figure 1a). The finetuned models have overall low perplexity. Models trained on earlier time slices show a strong preference for data from their respective time slice, whereas those trained on later slices do not show such prominent specialization.

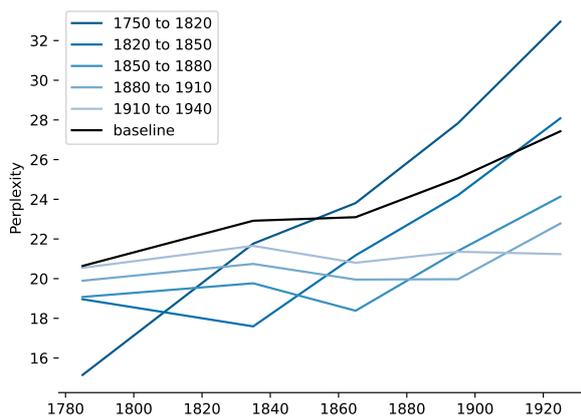
The scratch models are less fluent, but are specialized to their historical period. The scratch models uniformly produce their lowest perplexities when measured against their period’s corresponding test set. Cross-evaluation of the models against the other reserved test sets (Figure 1b) yields encouraging signs that the linguistic information captured by the scratch models follows an appropriate historical arc, and that there is little to no information from any of the extraneous time slices. Perplexity is low in the relevant period and increases monotonically both when testing older and newer texts, a sign of period-specific adaptation unique to the scratch models.

The scratch models perform reasonably on BLiMP despite generally underperforming the

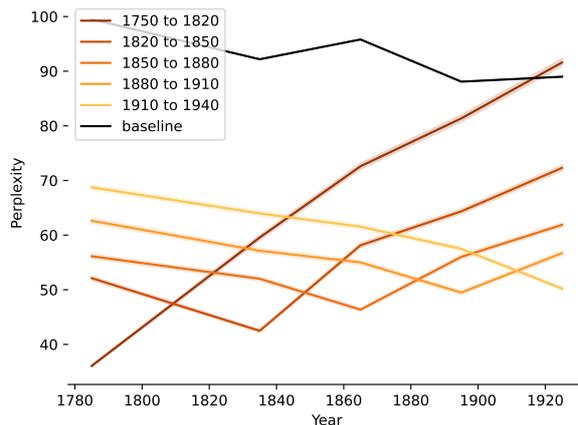
unadapted and finetuned models. The scratch models earn accuracy scores ranging from 0.70 to 0.74 (Table 3) on the maximally filtered version of the BLiMP evaluation set. For comparison, The two unadapted models earn aggregate scores of 0.74 (BabyLlama-2) and 0.82 (Llama3-8B) on the same set of BLiMP tasks. We expect this slight to moderate underperformance in both cases, as scratch was trained over more specific mix of 10M tokens than unadapted BabyLlama-2 and far fewer tokens than unadapted Llama3-8B. Nonetheless, scratch approaches the general competence of unadapted BabyLlama-2, and meets it in the later slices. The finetuned models consistently outperform the scratch models (Table 3). Beyond verifying that the models are usable models of language, we care about the contrastive differences between them. We note an increase in BLiMP competency over time. Given that BLiMP’s test samples are rendered in modern English, this demonstrates that our pretraining approach is potentially sensitive to diachronic grammatical change.

The scratch models respect diachronic lexical change in a way the finetuned models do not. While the finetuned models generally "succeed" more in completing a given cloze sentence drawn from OED (Figure 2), they achieve this in part by integrating information that defies the arrow of time, rendering them less useful for contrasting corpora and studying change. The leakage calculations collected in Figure 3 demonstrate that the finetuned models consistently overperform on future senses. Dividing model leakage by recall (Figure 4) shows that the finetuned models performance on future time slices remains inappropriately high when correcting for the performance of the scratch models. For details on cloze performance, see Appendix F.

Performing error analysis over each slice of both the scratch and finetuned models reinforces the conclusion that finetuned models from early timeslices perform inappropriately well on future cloze tasks, while the scratch models do not. Table 4 presents two examples of this phenomenon for models trained on the 1750-1820 timeslice. The first example, "pound" is ranked as most likely by the finetuned model and as highly unlikely by the scratch model. This car-centric sense cannot be in the training data for the 1750-1820 slice, as it predates the practical internal combustion engine. Thus, it is probable that the finetuned model is influenced by modern information it ingested in pretraining, resulting in "pound" being ranked as



(a) Finetuned models with unadapted Llama3-8B baseline



(b) Scratch models with unadapted BabyLlama-2 baseline

Figure 1: Cross-time perplexities

Model	1750-1820	1820-50	1850-80	1880-1910	1910-40
Scratch	0.70 ± 0.00	0.71 ± 0.00	0.73 ± 0.01	0.74 ± 0.00	0.74 ± 0.00
Finetuned	0.80	0.81	0.83	0.84	0.84

Table 3: Aggregate maximally filtered BLiMP accuracy across all timeslices.

inappropriately likely.

The second example, "silver" as elliptical for "silver medal," is completed by finetuned but also ranked as reasonably likely by scratch. However, this example includes a collocation of "gold," that may steer a model towards a higher probability of "silver." This implies a trend. While the scratch models can perform better than they "should", this likely only occurs when there is a collocated context clue. In contrast, the finetuned model performs out of its bounds on samples like "pound" that lack such clues. This, in combination with the perplexity, leakage and BLiMP evaluations, leads us to conclude that the **scratch models represent a better tradeoff between fluency and historical accuracy than the finetuned set.**

5.2 Exploratory analysis

The scratch models enable novel hypotheses about lexical changes across our corpus. One such approach takes advantage of a type of seeming diachronic error that differs from those explored above. Table 5 contains an example of this type of error, in the form of a cloze task centering on the phrase "end of the line." Both scratch and finetuned rank the correct completion within the top 20 possibilities. However, while we cannot rule out that the finetuned model is achieving accuracy due to its future knowledge (i.e. it has seen this

phrase in pretraining), we can do so for its scratch counterpart (a collocation search of the training data reveals that this exact construction is never used). Additionally, unlike the second type of error ("gold...silver") nothing in the context sentence makes "line" a likely conclusion by other direct means. Thus, the surprising performance of the scratch model is plausibly explained as a **prefiguration of a construction to come.** The way "line" is used in the 1750-1820 slice of the corpus predicts its ability to be used in this particular construction in the future. Examining the training corpus reveals numerous uses of "line" in hereditary (i.e. "end of one's line") writing (i.e. "the line ended") and martial (i.e. "the British line") contexts, all uses logically associated with the action of "ending." Some uses, especially those related to writing, seem sufficient to support this construction. In a sense, these uses "pave the way" for "end of the line." The restricted training of the scratch models allows them to capture this possible developmental narrative in a way the finetuned models cannot.

Employing the full diachronic battery of scratch models further demonstrates our method's ability to **detect subtle information about use from the top down.** Table 6 shows the ranking of the correct completion word ("cholera") for each of the models when completing the OED cloze:

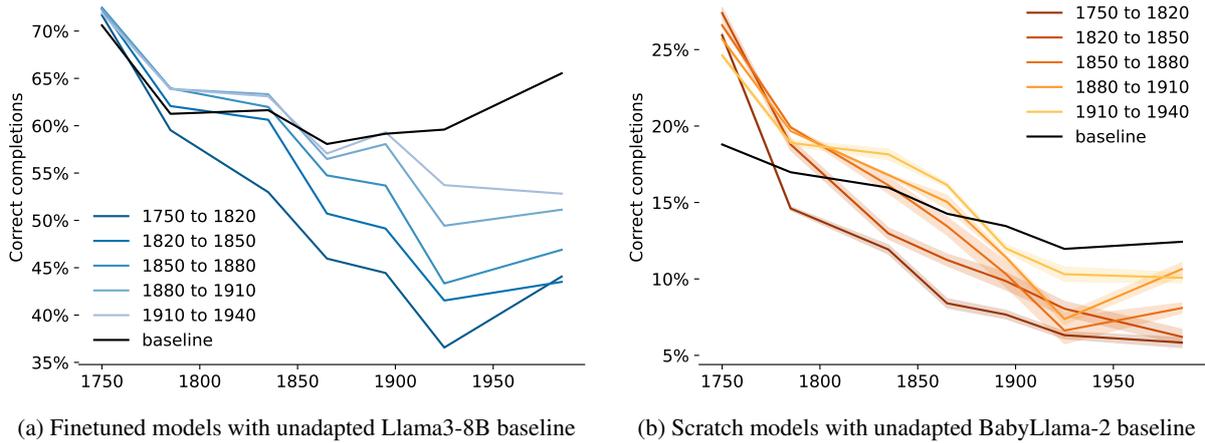


Figure 2: Model performance on the top 100 completion cloze task

Sentence	Definition	Year	Scratch	Finetuned
I'm going to sell my car... No more police towing [it] ..to a car pound .	A place in which vehicles impounded by the police or other authorities are kept...	1970	101±0.00	0
Hill ... which won three gold and a silver .	Elliptical for silver medal n.	1960	11.67±5.86	0

Table 4: Two examples for time slice 1750-1820 with their rank per model. Higher = more historically accurate. Rank 101 indicates the correct word is outside of the top $k = 100$. Rank 0 represents the models top choice.

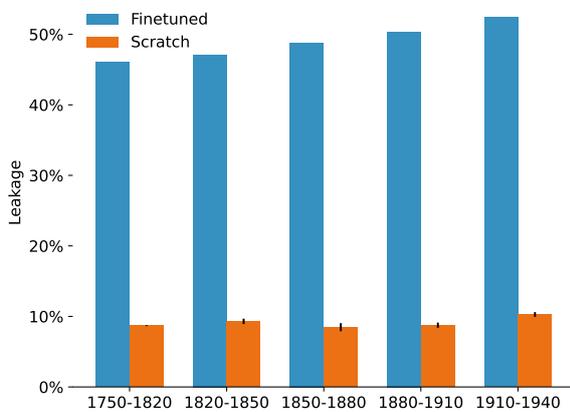


Figure 3: Leakage probability over scratch and finetuned models

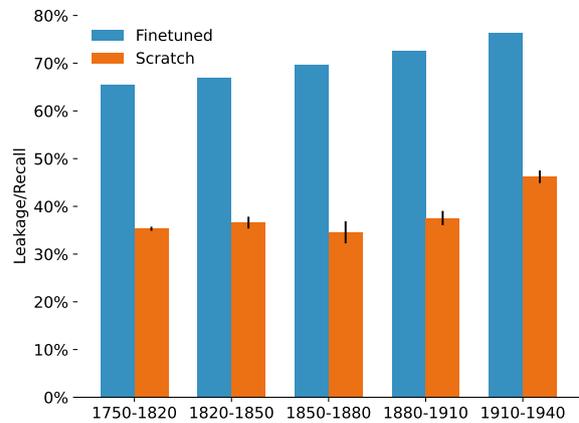


Figure 4: Leakage probability corrected by recall

The potatoes failed, the pigs were affected with a disease which the people called **cholera**

In modern usage, "cholera" denotes the illness associated with the pathogenic bacteria *Vibrio cholerae*, known historically as "asiatic cholera." Prior to the development of germ theory and the arrival of *Cholera vibrio* on Western shores,

"cholera," often paired with an additional identifier like "morbus" or "infantus," was used to describe a variety of intestinal maladies that shared similar symptoms (Rousseau and Haycock, 2003) (Barua, 1992).

However, this particular 1837 sense of "cholera" concerns a pig disease that presumably earned its appellation due to its gastric manifestation. The scratch sets' acceptance of this usage follows a dis-

Sentence	Definition	Year
They have nowhere to go. This is—how do the Americans say it?—the end of the line .	V. A direction or course of movement. the end of the line (transferred and figurative). Cf. the end of the road at end n..	1948

Table 5: The new sense of "line" is accepted by the finetuned (rank #1) and scratch (#14) models trained over the 1750-1820 timeslice

	1750-1820	1820-1850	1850-1880	1880-1910	1910-1940
Scratch	12±4.36	101±0.00	48.67±15.95	78.67±9.50	101±0.00
Finetuned	18	19	11	14	11

Table 6: Rank of "cholera" completion. Unadapted Llama3-8B ranks it 8, unadapted BabyLlama-2 ranks it 57. Rank 101 indicates it is outside the top $k = 100$.

tinct trajectory. The earliest model finds the usage most acceptable, the next few models vacillate, and the final (post-germ theory) model rejects it. In contrast, the finetuned models rank it as highly acceptable across the whole period. In doing so, they miss a less-explored facet of cholera’s conceptual solidification. Further research reveals that before the term’s conceptual solidification around Cholera vibrio, pig "choleras" shared semantic space with their human cousins (Cole and Brooks, 1961). Far from being a minor detail, this helps reveal how "cholera" became less a description of a set symptoms common to the animal kingdom and more a term for a particular human pathogen.

Examining the collocation of "cholera" in the training data of the earliest slice reveals that multiple forms of cholera, including "morbus" and "infantus," share lexical space with descriptions of their symptomatic overlap ("fever", "diarrhea" etc.). In the next slices, the term vacillates between these uses and the emerging understanding of cholera as a specific human disease capable of producing mass illness events, demonstrated by collocations like "epidemic" and "plague." Finally, it settles on its modern meaning (including collocation with "germ") in the last slice. (See supplementary materials). The scratch models capture and characterize this moment of conceptual solidification, hinting at their utility for investigating the details of historical social and knowledge changes from the top-down.

Discovering sense trajectories of interest from the bottom-up. Contrasting information provided by each scratch timeslice model allows more flexible automated hypotheses generation than the cloze approach utilized above. By taking this approach,

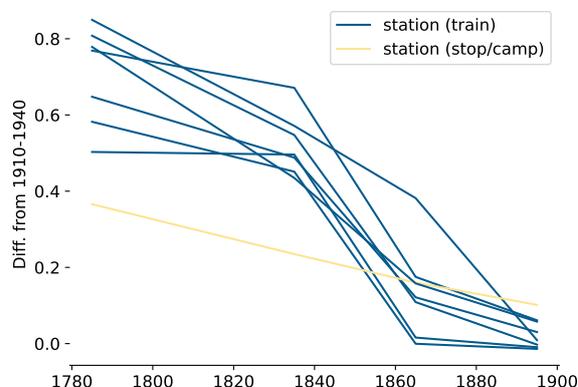


Figure 5: Natural appearances of "station" with a descending probability trajectory and manually labeled for sense.

word senses can be said to have trajectories across our corpus periods, as judged by their scratch model acceptability at each slice. For example, one would expect earlier models to be perplexed by the word "car", in the sense of an automobile, and expect the later models would accept it. To track how this shift occurs, we set the 1910-1940 model’s normalized per-word perplexity scores as a baseline, and retrieve all words that have a continually decreasing perplexity difference across time. For tractability, we subset this group to those with the largest change in acceptability between the first and last models.

This approach captures distinctions in usage over time, and separates synchronically distinct senses of words. Figure 5 depicts the trajectories for one of these words – "station." (See the full data in the supplementary materials.) Two senses of the word emerge after applying our filtering approach; one

is associated with a railroad station, and one with a stopover or encampment site. While both of these senses becomes more acceptable as time goes on, they follow distinct trajectories. The rail-related sense becomes precipitously more acceptable in the 1820-1850 slice, likely due to the adoption of rail technology during that period. In contrast, the camp/stopover sense begins its trajectory from a place of relative acceptability, and then proceeds to become smoothly more acceptable as time passes.

These observations lead to any number of hypotheses about the interaction between these senses that could be pursued by further means over larger corpora. For example, this information allows the question of whether the already-acceptable usage of "station" as camp or stop grew more acceptable due to the influence of the emerging rail-related sense. We offer some further analytical directions in Appendix G.

6 Conclusion and future work

Our approach leverages efficient pretraining in order to offer a novel, boundary-guaranteed, form of linguistic hypothesis discovery across comparative corpora. We demonstrate that the tradeoff of some model fluency for boundary accuracy pays off by enabling a series of exploratory techniques. While we believe this approach has potential for the specific use case examined above, diachronic change, we also believe that it is extensible, and can be leveraged in a similar way across different corpus divisions and fields. Further work could verify these beliefs by testing our approach's ability to detect linguistic shifts across synchronic boundaries. Additionally, increasing the size of the training corpora and developing historically-situated post-training regimes could lead to increased model capability, allowing for the discovery of knowledge-level hypotheses relevant to disciplines like literary studies and history.

7 Limitations

This work has a few primary limitations. The first is selection of data. While we were able to attribute text publication dates with reasonable accuracy, it is not clear that our method will work on less well-documented texts. This had the effect of limiting our training corpora for this study to works like long-form fiction and nonfiction that are heavily commented on in the historical record. The second is model performance. While our scratch models

are reasonably fluent, the tradeoff between performance and historical certainty they traffic in could potentially be further minimized by other efficient training techniques. Finally, we chose to limit our training data to a single language and modality for this study.

References

- Gati Aher, RosaI. Arriaga, and A. Kalai. 2022. [Using large language models to simulate multiple humans and replicate human subject studies](#). In *International Conference on Machine Learning*.
- Dhiman Barua. 1992. History of cholera. In *Cholera*, pages 1–36. Springer.
- Christin Beck and Marisa Köllner. 2023. Ghisbert-training bert from scratch for lexical semantic investigations across historical german language stages. In *Proceedings of the 4th Workshop on Computational Approaches to Historical Language Change*, pages 33–45.
- Younes Belkada, Tim Dettmers, Artidoro Pagnoni, Sylvain Gugger, and Sourab Mangrulkar. 2023. Making llms even more accessible with bitsandbytes, 4-bit quantization and qlora.
- Jeffrey Cheng, Marc Marone, Orion Weller, Dawn Lawrie, Daniel Khashabi, and Benjamin Van Durme. 2024. Dated data: Tracing knowledge cutoffs in large language models. *arXiv preprint arXiv:2403.12958*.
- Leshem Choshen, Ryan Cotterell, Michael Y. Hu, Tal Linzen, Aaron Mueller, Candace Ross, Alex Warstadt, Ethan Wilcox, Adina Williams, and Chengxu Zhuang. 2024. [\[call for papers\] the 2nd BabyLM Challenge: Sample-efficient pretraining on a developmentally plausible corpus](#). *Computing Research Repository*, arXiv:2404.06214.
- Clarence Gordon Cole and Stanley N Brooks. 1961. *History of Hog Cholera Research in the US Dept. of Agriculture, 1884-1960*. 240-241. Agricultural Research Service, US Department of Agriculture.
- Bhuvan Dhingra, Jeremy R Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.
- Felix Drinkall, Eghbal Rahimikia, Janet Pierrehumbert, and Stefan Zohren. 2024. [Time machine GPT](#). In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3281–3292, Mexico City, Mexico. Association for Computational Linguistics.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence

- Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonnell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. [A framework for few-shot language model evaluation](#).
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Thora Hagen. 2025. [Lexical semantic change annotation with large language models](#). In *Proceedings of the 9th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2025)*, pages 172–178, Albuquerque, New Mexico. Association for Computational Linguistics.
- Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.
- Kasra Hosseini, Kaspar Beelen, Giovanni Colavizza, and Mariona Coll Ardanuy. 2021. Neural language models for nineteenth-century english. *arXiv preprint arXiv:2105.11321*.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.
- Michael Y. Hu, Aaron Mueller, Candace Ross, Adina Williams, Tal Linzen, Chengxu Zhuang, Leshem Choshen, Ryan Cotterell, Alex Warstadt, and Ethan Gotlieb Wilcox, editors. 2024. *The 2nd BabyLM Challenge at the 28th Conference on Computational Natural Language Learning*. Association for Computational Linguistics, Miami, FL, USA.
- Maxim Ifergan, Leshem Choshen, Roei Aharoni, Idan Szpektor, and Omri Abend. 2024. Beneath the surface of consistency: Exploring cross-lingual knowledge representation sharing in llms. *arXiv preprint arXiv:2408.10646*.
- Anna A Ivanova, Aalok Sathe, Benjamin Lipkin, Unnathi Kumar, Setayesh Radkani, Thomas H Clark, Carina Kauf, Jennifer Hu, RT Pramod, Gabriel Grand, et al. 2024. Elements of world knowledge (ewok): A cognition-inspired framework for evaluating basic world knowledge in language models. *arXiv preprint arXiv:2405.09605*.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.
- Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*.
- Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning*.
- Enrique Manjavacas and Lauren Fonteyn. 2022. Adapting vs. pre-training language models for historical languages. *Journal of Data Mining & Digital Humanities*, (Digital humanities in languages).
- Francesco Periti, David Alfter, and Nina Tahmasebi. 2024a. [Automatically generated definitions and their utility for modeling word meaning](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14008–14026, Miami, Florida, USA. Association for Computational Linguistics.
- Francesco Periti, Pierluigi Cassotti, Haim Dubossarsky, and Nina Tahmasebi. 2024b. [Analyzing semantic change through lexical replacements](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4495–4510, Bangkok, Thailand. Association for Computational Linguistics.
- Francesco Periti, Haim Dubossarsky, and Nina Tahmasebi. 2024c. [\(chat\)GPT v BERT dawn of justice for semantic change detection](#). In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 420–436, St. Julian's, Malta. Association for Computational Linguistics.
- Francesco Periti and Stefano Montanelli. 2024. Lexical semantic change through large language models: a survey. *ACM Computing Surveys*, 56(11):1–38.
- Bhawna Piryani, Jamshid Mozafari, and Adam Jatowt. 2024. Chroniclinaamericaqa: A large-scale question answering dataset based on historical american newspaper pages. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2038–2048.
- Felipe Maia Polo, Seamus Somerstep, Leshem Choshen, Yuekai Sun, and Mikhail Yurochkin. 2024. Sloth: scaling laws for llm skills to predict multi-benchmark performance across families. *arXiv preprint arXiv:2412.06540*.
- Wenjun Qiu and Yang Xu. 2022. Histbert: A pre-trained language model for diachronic lexical semantic analysis. *arXiv preprint arXiv:2202.03612*.
- George S Rousseau and David Boyd Haycock. 2003. Coleridge's cholera: Cholera morbus, asiatic cholera, and dysentery in early nineteenth-century england. *Bulletin of the History of Medicine*, 77(2):298–331.

- Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. [SemEval-2020 task 1: Unsupervised lexical semantic change detection](#). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1–23, Barcelona (online). International Committee for Computational Linguistics.
- Jean-Loup Tastet and Inar Timiryasov. 2024. [BabyLlama-2: Ensemble-distilled models consistently outperform teachers with limited data](#). In *The 2nd BabyLM Challenge at the 28th Conference on Computational Natural Language Learning*, pages 292–301, Miami, FL, USA. Association for Computational Linguistics.
- Sean Trott. 2024. [Can large language models help augment english psycholinguistic datasets?](#) *Behavior Research Methods*, 56:6082 – 6100.
- Khonzoda Umarova, Lillian Lee, and Laerdon Kim. 2025. [Current semantic-change quantification methods struggle with discovery in the wild](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 35342–35355, Suzhou, China. Association for Computational Linguistics.
- Ted Underwood, Laura K Nelson, and Matthew Wilkens. 2025. [Can language models represent the past without anachronism?](#) *arXiv preprint arXiv:2505.00030*.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. [GLUE: A multi-task benchmark and analysis platform for natural language understanding](#). In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.
- Alex Warstadt and Samuel R Bowman. 2022. [What artificial neural networks can tell us about human language acquisition](#). In *Algebraic structures in natural language*, pages 17–60. CRC Press.
- Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, and Ryan Cotterell. 2023. [Findings of the BabyLM challenge: Sample-efficient pretraining on developmentally plausible corpora](#). In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 1–34, Singapore. Association for Computational Linguistics.
- Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. [BLiMP: The benchmark of linguistic minimal pairs for English](#). *Transactions of the Association for Computational Linguistics*, 8:377–392.
- Ethan Gotlieb Wilcox, Jon Gauthier, Jennifer Hu, Peng Qian, and R. Levy. 2020. [On the predictive power of neural language models for human real-time comprehension behavior](#). *ArXiv*, abs/2006.01912.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

A Additional model and training details

DoRa adapters of rank 16 were chosen for efficiency purposes. We adopt the hyperparameters used in the original paper (Table 7).

Hyperparameters (DoRA)	LLaMA3-8B
Rank r	16
α	32
Dropout	0.05
Optimizer	AdamW
LR	1×10^{-4}
LR Scheduler	Linear
Batch size	16
Warmup Steps	100
Epochs	3
Where	Q, K, V, Up, Down

Table 7: Hyperparameter configurations of DoRa for LLaMA3-8B.

We finetune using next-token prediction loss on the data slices. We train the models for three epochs, with one training run taking around eight hours on a single A100 GPU, which emits 0.11KG CO_2 an hour (Lacoste et al., 2019). We choose the best model checkpoints based on evaluation loss. Notably, models trained on data after 1850 reached optimal performance after a single epoch, while those trained over earlier periods continued to show improvement during the second epoch.

We train the scratch models using the BabyLlama-2 training recipe. We adopt the same Llama-345M model (Table 9) and training hyperparameters (Table 8) as the original paper.

BabyLlama-2 uses a distillation strategy where the logits of two trainer models are used to train a student model. Notably, the teacher and student models are of the same size. We initialize a Byte-Pair-Encoding tokenizer for each time slice and

Hyperparameter	Value
Learning rate	$7 \cdot 10^{-4}$
Number of epochs	8
Batch size	128
Weight decay	5
Distillation α	0.5

Table 8: Training and distillation hyperparameters of BabyLlama-2

Hyperparameter	Value
Vocabulary size	16,000
Number of layers	32
Number of heads	15
Number of KV heads	5
Embedding dimension	960
Hidden dimension	2560
Total parameters	345M

Table 9: BabyLlama-2 Model Architecture.

train two teacher models over the training data for eight epochs. We select the model with the best validation score (consistently epoch four during our runs). Training a teacher model took around 32 minutes on a single A100 GPU. From the two teachers, we then distill a student model using the distillation loss after [Hinton et al. \(2015\)](#), with $L = \alpha L_{CE} + (1 - \alpha)L_{KL}$. This loss is made up in equal parts of the normal next token prediction loss and the loss over the soft trainer logits. We train the student over eight epochs; the last epoch consistently having the lowest evaluation loss. Training the student model took 3 hours and 20 minutes on an A100.

B Attribution pipeline details

We extract from WD all entities with an occupation of "author" or "writer" that also have birth dates that fall within this range. We further constrain this subset by filtering it to only include authors WD indicates were known to write in English.

We then fuzzily match this set of authors to those in PGC. The first pass uses Levenshtein distance matching with a predefined threshold in combination with any extractable birth and death information to match PGC authors to the list sourced from WD. The optional second pass uses only fuzzy string matching with a stricter predefined thresh-

old, and matches any remaining PGC authors to an author from WD. This second pass allows for the inclusion of authors without WD-provided date information, compensating for the further loss in certainty with tighter regulation of name similarity. The result of this stage is a mapping between WD authors and PGC authors with an associated list of their works found in PG.

To validate open source and propriety LLM performance on work-date attribution we manually annotated a sample (n=1054) of known-author works with their date of writing using publication information sourced from internet repositories like the HathiTrust collection (This material is available in the supplement). We then used one open weight model (Llama3.3-70B quantized to 4 bits) and two proprietary models (GPT-4, GPT-4o) to zero-shot attribute the dates of works using the following prompt:

When was the work {} by {} written?
Answer just with the year.

Where the first {} was replaced with the work title and the second {} by the work author. We then evaluated performance with a tolerance of +/- 1 year to account for the historically common practice of assigning publication date to copyright year. Noting systematic error in the results provided by the best performing model at this stage (GPT-4o) we collected the set of erroneously attributed texts produced by this model and undertook another round of hand annotation on this set, spending additional effort to source historical materials (publishing industry trade journals, library records) that could disambiguate questionable attributions or provide evidence of earlier publications not in the digitized record. We then re-evaluated the models with tolerances of +/- 1 and 10 years, allowing a match to either date attribution to be acceptable. Additionally, we evaluated the models after disqualifying scores with extreme difference (+/- 50 years) from their ground score, to assess the impact of having a more certain source of information (say, author birth and death dates) that pre-restricts correct answers to a tighter range. Table 10 shows that while the closed-source models perform the best under these conditions, the open source model performs well enough to serve as a first point of departure.

Notably, this approach is flexible – broader diachronic slices justify tolerating more variance.

	+/-1	+/-10	DQ +/-1	DQ +/-10
Llama3.3	0.63	0.81	0.70	0.88
GPT-4	0.74	0.89	0.87	0.99
GPT-4o	0.82	0.84	0.96	0.94

Table 10: Performance on work date attribution per LLM. +/- indicates year delta tolerance threshold, DQ indicates that extreme variations from the ground scores (+/-50) were not considered

C Cloze evaluation set details

The cloze evaluation set contains 50.4 thousand examples. 14.6 thousand examples remain after filtering, a large portion of which is of old english origin as can be seen in Figure 6. Evaluation was performed over the top 100 word completion task. If the word appeared within the top 100 words (case insensitive) the completion was considered successful. For evaluation the senses were grouped by time slice. In Figures 2 and 9, each model was evaluated over each time slice. In the leakage reports (Figures 3 and 4) the model performance was contrasted between the senses created before and after the models respective training cutoff.

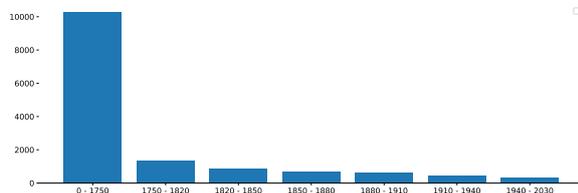


Figure 6: Count of cloze tasks for per time slice for the set filtered for our data (14.6 thousand examples).

Band	Freq./mil.	% in OED
8	>1,000	0.02%
7	100 – 1,000	0.18%
6	10 – 100	1%
5	1 – 10	4%
4	0.1 – 1	11%
3	0.01 – 0.1	20%
2	<0.01	45%
1	–	18%

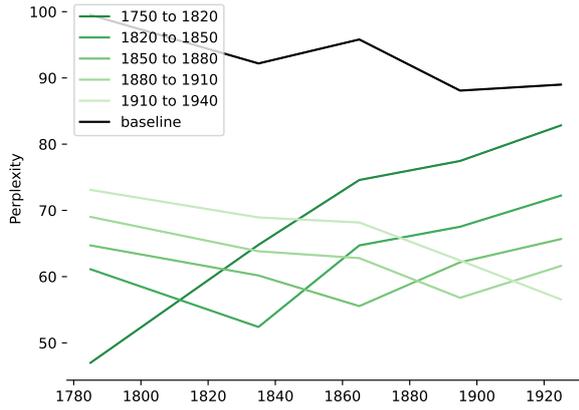
Table 11: Word Frequency Bands and their respective counts per a million words and the percentage of non-obsolete OED entries

D BLIMP evaluation set details

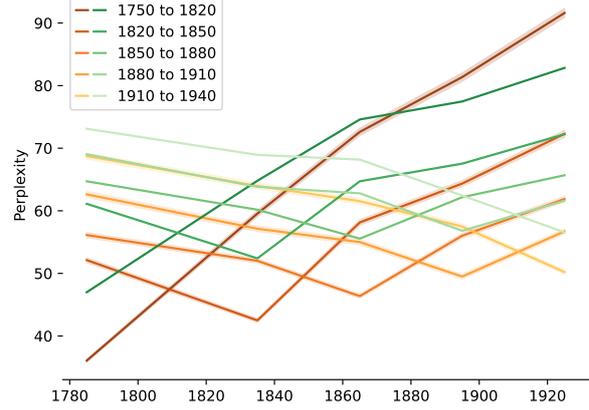
The BLIMP task was filtered from the training data across all time slices, resulting in an uneven distribution over BLIMP subtasks, an overview of this can be seen in Table 12.

E Performance of DoRA finetuned pretrained BabyLlama-2 (small)

As an additional point of comparison, we also finetuned the distribution checkpoint of BabyLlama-2 using the data from our historical timeslices (*'finetuned small'*). This checkpoint of BabyLlama-2, provided by the authors of the BabyLlama-2 paper, was pretrained on the BabyLM mixture of 10 million tokens for 8 epochs. While this comparison is not a perfect one, as the *'finetuned small'* model has prior access to more linguistic information from its pretraining, the resulting small models under-perform the approaches explored in the main body of the paper (Figure 7). They also suffer from more leakage than scratch (Figure 8).

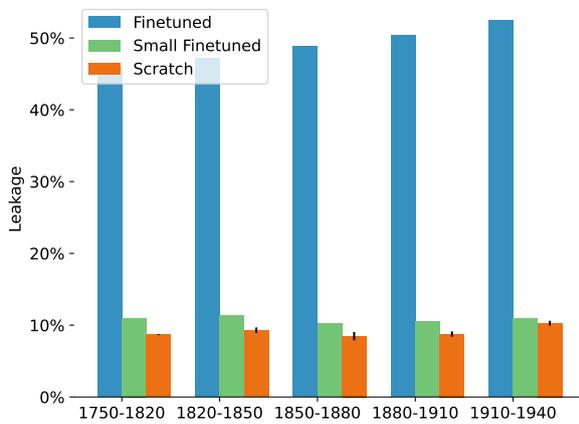


(a) Perplexity of small finetuned models with baseline.

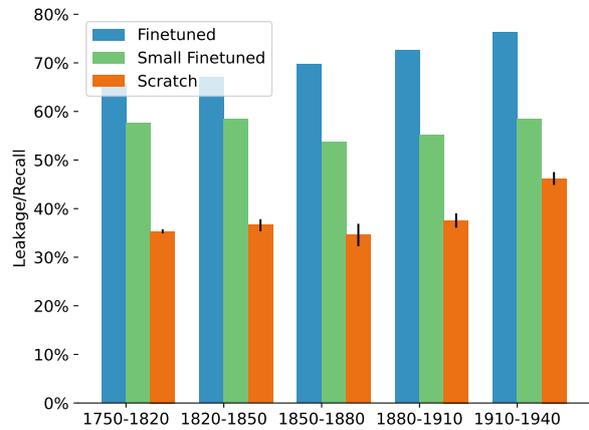


(b) Small finetuned model compared to scratch.

Figure 7: Cross-time perplexity comparison of small and scratch



(a) Leakage probability over scratch and finetuned models.



(b) Leakage probability corrected by recall.

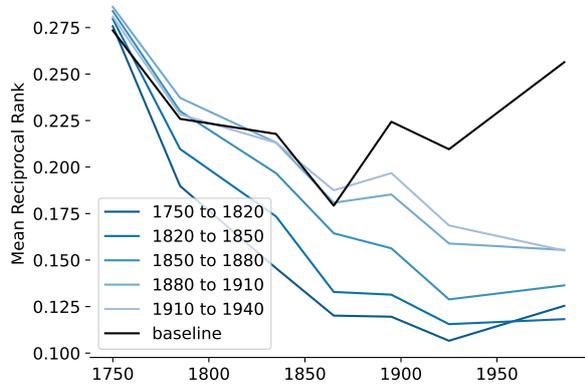
Figure 8: Leakage charts including the small finetuned models.

F Additional model performance information

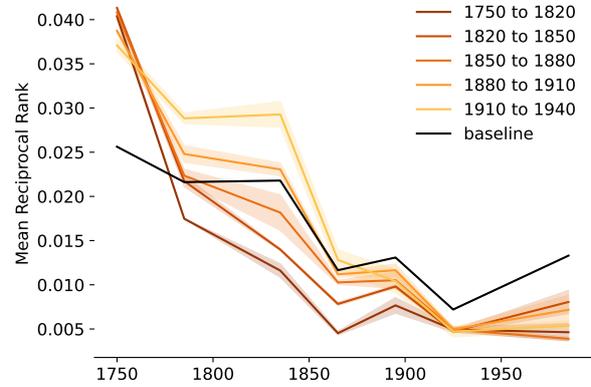
We include an overview of mean reciprocal rank over time, for a more detailed insight into model performance (Figure 9).

G Further analytical commentary

In a second, more cumulative analysis (Figure 10), words with consistently high perplexity differences were highlighted. The underlying reason for these variations is varied. Some words show semantic shifts, such as “car” (automobile) “plane” (airplane) and “inspector” (detective), while others are a part of novel word combinations, which had gained popularity such as “skirt” in the context of “hobble skirt” or “Victoria” in the context “Queen Victoria”. While these insights cannot be pinpointed to a single phenomenon, they offer valuable insights into the training corpora.



(a) Finetuned MRR with unadapted Llama3 baseline



(b) Scratch MRR with unadapted BabyLlama-2 baseline

Figure 9: MRR performance on the cloze task

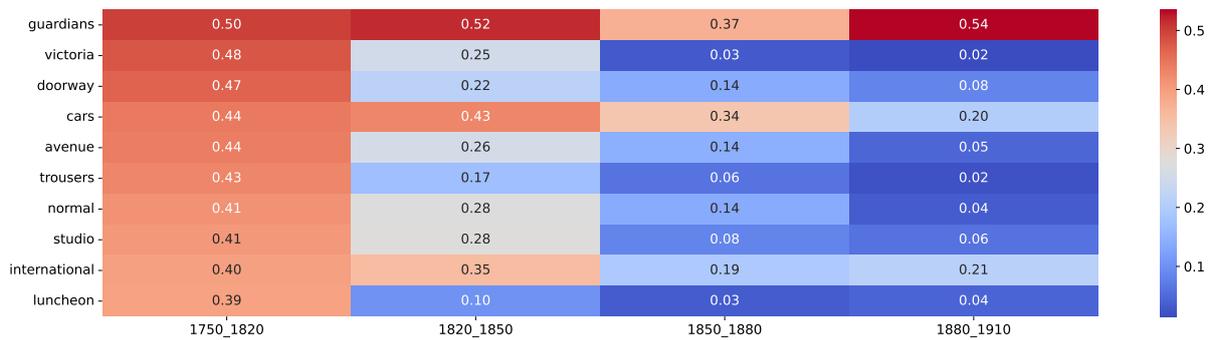


Figure 10: Cumulative perplexity results.

Task	Filtered	Original
blimp_principle_A_domain_2	322	1000
blimp_ellipsis_n_bar_2	199	1000
blimp_tough_vs_raising_1	501	1000
blimp_animate_subject_passive	334	1000
blimp_irregular_past_participle_adjectives	658	1000
blimp_determiner_noun_agreement_2	457	1000
blimp_wh_vs_that_with_gap_long_distance	364	1000
blimp_sentential_negation_npi_licensor_present	478	1000
blimp_wh_vs_that_with_gap	486	1000
blimp_determiner_noun_agreement_with_adj_irregular_2	390	1000
blimp_irregular_past_participle_verbs	522	1000
blimp_coordinate_structure_constraint_object_extraction	377	1000
blimp_anaphor_number_agreement	603	1000
blimp_wh_questions_subject_gap	268	1000
blimp_principle_A_domain_1	346	1000
blimp_drop_argument	468	1000
blimp_determiner_noun_agreement_with_adjective_1	438	1000
blimp_complex_NP_island	240	1000
blimp_principle_A_domain_3	304	1000
blimp_matrix_question_npi_licensor_present	486	1000
blimp_left_branch_island_simple_question	467	1000
blimp_expletive_it_object_raising	172	1000
blimp_wh_questions_subject_gap_long_distance	166	1000
blimp_causative	287	1000
blimp_determiner_noun_agreement_with_adj_irregular_1	300	1000
blimp_wh_questions_object_gap	248	1000
blimp_regular_plural_subject_verb_agreement_1	314	1000
blimp_determiner_noun_agreement_with_adj_2	423	1000
blimp_existential_there_object_raising	193	1000
blimp_inchoative	506	1000
blimp_superlative_quantifiers_1	616	1000
blimp_anaphor_gender_agreement	516	1000
blimp_tough_vs_raising_2	333	1000
blimp_regular_plural_subject_verb_agreement_2	511	1000
blimp_determiner_noun_agreement_irregular_2	391	1000
blimp_principle_A_case_2	369	1000
blimp_superlative_quantifiers_2	614	1000
blimp_existential_there_subject_raising	398	1000
blimp_npi_present_2	480	1000
blimp_distractor_agreement_relational_noun	237	1000
blimp_wh_island	487	1000
blimp_irregular_plural_subject_verb_agreement_1	289	1000
blimp_transitive	263	1000
blimp_sentential_subject_island	372	1000
blimp_principle_A_c_command	427	1000
blimp_principle_A_case_1	333	1000
blimp_existential_there_quantifiers_1	410	1000
blimp_principle_A_reconstruction	571	1000
blimp_existential_there_quantifiers_2	367	1000
blimp_wh_vs_that_no_gap_long_distance	217	1000
blimp_adjunct_island	305	1000
blimp_animate_subject_trans	269	1000
blimp_distractor_agreement_relative_clause	291	1000
blimp_coordinate_structure_constraint_complex_left_branch	266	1000
blimp_wh_vs_that_no_gap	290	1000
blimp_only_npi_scope	272	1000
blimp_irregular_plural_subject_verb_agreement_2	503	1000
blimp_passive_1	235	1000
blimp_determiner_noun_agreement_1	427	1000
blimp_passive_2	381	1000
blimp_only_npi_licensor_present	447	1000
blimp_npi_present_1	470	1000
blimp_ellipsis_n_bar_1	214	1000
blimp_left_branch_island_echo_question	439	1000
blimp_determiner_noun_agreement_irregular_1	283	1000
blimp_sentential_negation_npi_scope	269	1000
blimp_intransitive	494	1000
Total	25373	67000

Table 12: Overview BLIMP filtered.