On the Effect of Hyperparameters in Language Modeling for Computational Linguistics

by

Ruoxi Ning

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2026

© Ruoxi Ning 2026

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contribution

Ruoxi Ning was the sole author for Chapters 1, 2, and 7, which were written under the supervision of Dr. Freda Shi and were not written for publication.

This thesis consists in part of one manuscript written for publication. Exception to the sole authorship of material is as follows:

Research presented in Chapter 3, 4, 5, and 6:

Dr. Freda Shi is a faculty member at the University of Waterloo and the Vector Institute, who supported conducting this work. Yongpeng Zhu, Qingcheng Zeng, and Dr. Tatsuki Kuribayashi are co-authors on any publications relating to this work. This research was conducted at the University of Waterloo by Ruoxi Ning under the supervision of Dr. Freda Shi. Ruoxi Ning wrote the draft manuscript, which all co-authors contributed intellectual input on. Yongpeng Zhu was the primary coder for Chapter 3, participated in coding and analysis. Qingcheng Zeng and Dr. Tatsuki Kuribayashi contributed to the idea discussion.

As lead author of this chapter, I was responsible for contributing to conceptualizing study design, carrying out data collection and analysis, and drafting and submitting manuscripts. My coauthors provided guidance during each step of the research and provided feedback on draft manuscripts.

Abstract

Experimental linguistics serves as a vital resource and evaluation criterion for natural language understanding in contemporary machine learning. While linguistically motivated experiments often use controlled settings to explore how models learn, they frequently overlook the role of training hyperparameters as a source of randomness. In this study, by replicating three representative experiments [14, 41, 47] with hyperparameters varied over a practical range, we show that modest hyperparameter changes can alter qualitative conclusions about models' linguistic abilities or even reverse cross-model comparisons, while other findings remain stable across the explored space. These results suggest that some prior studies in this field may partly reflect optimization artifacts rather than the inductive bias of a model class. We argue that hyperparameter sensitivity is an under-examined factor that can meaningfully influence model behaviors, and thus call for future linguistically motivated experiments with neural models to attend to hyperparameter selection and the stability of the findings across the configuration space.

Acknowledgments

This thesis marks a temporary milestone in my scientific journey, and I hope it is far from the end. Many people have shaped and supported me along the way, and this page cannot be an exhaustive list.

I am sincerely grateful to my advisor, Freda Shi, for selecting me for this project and giving me the opportunity to dedicate myself to learning and research with adequate funding. I cannot adequately express my appreciation for her inspiration in research topics that, for the first time, I found truly compelling; for her rigorous guidance in shaping my research and coding practices; and for her leadership in fostering a group that welcomes bold ideas and open discussion. I cannot imagine what my life would be like without this short but remarkable journey in the CompLING Lab.

This work was supported by funding from the University of Waterloo (through Freda), the Vector Institute, and a scholarship from Cohere. Without their support, completing this degree would have been difficult.

I am also grateful to the University of Waterloo for offering a wide range of courses in mathematics, machine learning theory, and AI, which broadened my perspective. Courses taught by Freda, Victor Zhong, Yuntian Deng, Pascal Poupart, Jesse Hoey, Shai Ben-David, and others have either inspired me at a fundamental level or enhanced my programming and formal proof skills, although for some I only had the chance to audit. My experience as a TA for CS 486/686 (Introduction to AI) and CS 245 (Logic and Computation) gave me valuable teaching experience and the opportunity to explore topics beyond my research that I nonetheless found deeply interesting.

I am thankful to my labmates for maintaining a supportive environment for discussion, where questions and ideas could freely circulate, even if many of them were not ultimately developed into practical research. Thanks to Hala Sheta for her wild and creative thoughts; Michael Ogezi for connecting me with schoolmates and maintaining our communication in French; Xiaoxi Luo for our wide-ranging discussions and her supportive presence; Kath Choi for discussions about career plans and our time spent climbing; Yixuan Wang for sharing knowledge about profiling toolkits; and Sheng Yao and Yifan Jiang for our discussions in linguistics. I also thank our extended group members, Wentao Zhang and Yuxuan Li, as well as other schoolmates, Benjamin Schineider and Luke Rivard, for sharing their research progress with me and for the time we spent hiking or at the gym.

Thanks to my remote friends and to the reading and discussion groups we held together. Special thanks to my former schoolmates Qingcheng Zeng, Kexun Zhang, Zi Ye, Xingyuan Zhao,

and Yiheng Yang, who have been sources of motivation in my love for and pursuit of knowledge in computational linguistics.

I thank my mother and father for their support, and I congratulate my mother, who recently became Dr. Du with research fields in nursing and Chinese medicine at the age of 48. Given where she started, she is a heroine to me. Their determination will be a constant source of encouragement for me to move forward.

Land Acknowledgment

I acknowledge hereby that I work on the traditional territory of the Neutral, Anishinaabeg, and Haudenosaunee peoples. The main campus of University of Waterloo is situated on the Haldimand Tract, the land granted to the Six Nations that includes six miles on each side of the Grand River.

Dedication

To all the truths in the garden of forking paths.

Table of Contents

Author's Declaration ii

Statement of Contribution iii

Abstract iv

Acknowledgments v

Land Acknowledgment vii

Dedication viii

List of Figures xii

List of Tables xiv

1 Introduction 1

2 Background 3

2.1 Language Modeling 3

2.2 Linguistic Competence in Behavioral Experiments 4

2.3 Replication Crisis in Machine Learning 4

2.4 Hyperparameter Optimization 5

3 Experiment A: Distributive Word Learning Pattern **6**
3.1 Training Procedure . . . 6
3.2 Claims to Be Examined . . . 7
3.3 Results across Configurations . . . 8
3.4 Discussion . . . 11

4 Experiment B: Syntactic Generalization Ability **13**
4.1 Training Setting . . . 14
4.2 Claims to Be Examined . . . 14
4.3 Results across Configurations . . . 15
4.4 Discussion . . . 17
4.5 Analysis on Effect of Tuning on SG Scores . . . 17
4.6 Analysis on Syntax Ability by Test Suite . . . 18

5 Experiment C: Word Order Preference **20**
5.1 Training Setting . . . 20
5.1.1 Claims to Be Examined . . . 21
5.2 Hypothesis and metrics. . . . 21
5.3 Results across Configurations . . . 23
5.4 Discussion . . . 24
5.5 Analysis on Word Order Preference by Switch . . . 24

6 Discussion **26**
6.1 Effects of Hyperparameters . . . 26
6.2 Effects of Early Stopping . . . 27
6.3 Positions on Future Research . . . 28
6.3.1 Hyperparameter Tuning . . . 28
6.3.2 Fragile Conclusion . . . 29
6.3.3 Reporting Standard . . . 29
6.3.4 Replication Crisis . . . 29
6.3.5 Exploratory and confirmatory data analysis . . . 30

7 Conclusion 31

References 32

APPENDICES 42

A Tuning Space 43

B Implementation Difference in Each Experiment 46

C P-value in [47] and change of the metric. 47

List of Figures

3.1 Learning curves of the word ‘eat’ for the LSTM model trained on each configuration. 9

3.2 Curves of KL-divergence between the real distributions and learned predictions during training. 10

4.1 Comparison of the original paper’s results, and the replicated test perplexities and SG scores for LSTM and GPT-2 on all configurations, each dot denoting one distinct training run. Red dots denote the original results. Same as [41], since GPT-2’s losses have been converted from token-count-based into word-count-based, the perplexities of GPT-2 will increase in numbers, and thus perplexities are not directly comparable among model architectures. 16

4.2 SG scores of each configuration, ranked by validation set perplexity on XS set. Each point represents the score delta relative to the average SG score across all models. Average validation-set perplexities are annotated over the bars. Error bars represent 95% confidence interval through bootstrapping. 18

4.3 SG scores per syntactic ability of each configuration, ranked by validation set perplexity on XS set. 19

6.1 The dynamics of test-set perplexities SG scores according to training procedure. Early stopping is decided by convergence on validation-set, with best checkpoint loaded. LSTM Config 9 misses a 1/4 XS point since the corresponding training instances converge within three epochs. 28

C.1 An example of poor correlation with high p-values in Pearson correlation testing: WALS frequency and model perplexity of grammars in Experiment C. The low correlation and high probability to reject the correlation demonstrated by these scatter plots hinder both drawing a conclusion and observing a variance resulting from hyperparameters. 48

C.2 In experiment C, the word order preference experiment, coefficients from fitted linear regression on the training results in the original paper and replication run. . 49

List of Tables

3.1 In experiment A, parameters taken from fitted sigmoid curves on tested configurations. 'Upper' and 'lower' indicate the max and min y-values on the curve. 'Xmid' denotes the x-value of the midpoint of the curve. 'Scale' is a factor describing the range of the x-axis. The greater the 'scale' is, the flatter the curve will be. 11

3.2 Coefficients in experiment A. Perplexity, significant predictors, and R^2 are reported in the original experiment and in the other four sets of configurations. Significant predictors are marked by asterisks ($p < 0.05$ *; $p < 0.01$ **; $p < 0.001$ ***). Signs of coefficients are notated in parentheses. The R^2 denotes the adjusted R^2 in a regression using all five predictors. The HP records the differences in hyperparameters between the corresponding setting and the base setting, with colors separating the model parameter and training parameter. 12

5.1 Estimated coefficients related to the original paper's conclusion for all configurations in Experiment C. Coefficients with no significance are denoted as '-'. Significance codes: '***' for $p < 0.001$, '**' for $p < 0.01$, and '*' for $p < 0.05$. 'Repl' denotes the replication experiment. All adjusted R^2's are above 0.9. The HP records the differences in hyperparameters between the corresponding setting and the base setting, with colors separating the model parameter and training parameter. 22

5.2 Test-set perplexities from replication of experiment C, comparing with results open-sourced by the original paper. 23

5.3 Estimated coefficients for all configurations on models' grammar-digit preferences in Experiment C. $\beta_{5\bullet}$ summarizes the domain-model interactions ($d_1, \ldots, d_7$ * model). Coefficients with no significance are denoted as '-'. Significance codes: '***' for p < 0.001, '**' for p < 0.01, and '*' for p < 0.05. 'Repl' denotes the replication experiment. All adjusted R^2's are above 0.9. The HP records the differences in hyperparameters between the corresponding setting and the base setting, with colors separating the model parameter and training parameter. . . . 25

5.4 Grammar definition of the artificial language adopted in the word order preference experiment. 25

6.1 Hyperparameters as significant predictors of SG scores, in LSTM model (up) and GPT-2 model (bottom). Signs determined by the sign of coefficients in the overall regression, and significance by the likelihood ratio test. 27

A.1 Configurations of all experiments performed. 44

A.2 Hyperparameter optimization space of experiment B during Bayes tuning. 45

Chapter 1

Introduction

Language models (LMs) have increasingly become a subject of linguistically motivated experiments on a wide range of topics from low-level fine-grained learning curves of grammar rules to high-level simulation of typological distribution of language structures [65, 84, 56, 87, *inter alia*]. One appeal of LM-based linguistic experiments lies in their ability to enable controlled ablation studies of learning scenarios [81] and to precisely track learning dynamics. Thus, a considerable portion of work involves training models from scratch under controlled conditions. These studies provided new perspectives on how models acquire and understand word and grammar rules, on cross-architecture comparisons of linguistic ability, and on comparisons between human and model cognition.

Such experiments can often be limited to demonstrating the proof of concept with an arbitrary model instance, or training without model tuning. However, it is commonly accepted in the machine learning domain that hyperparameters, including learning rate, initialization, optimization settings, and type of computation, affect the training dynamics and final behavior [48, 16, 22, 73, *inter alia*]. Meanwhile, the concept of reproducibility has been a guiding principle in machine learning [36, 5]. Much recent machine learning research cannot achieve solid reproducibility when performed with arbitrarily chosen hyperparameters [11]. Under-reporting or arbitrarily choosing hyperparameters will hinder the genuine observation of the model's behavior and the generalization the research claims, since the quantitative evaluation partly reflects the effect of hyperparameters.

In this thesis, we systematically examine how hyperparameter variation affects linguistically oriented findings in neural networks. We revisit three representative experiments on language models, spanning low-level word-learning curves, syntactic knowledge, and the typological distribution of linguistic features: (A) word acquisition patterns, (B) syntactic generalization ability,

and (C) word order preferences. We focus on linguistically driven studies that adopt behavioral (non-probing) methods to evaluate linguistic competence. This line of work, influenced by the cognitive revolution and the generative tradition, primarily targets morphological and syntactic phenomena, with comparatively little attention to other domains such as phonetics and pragmatics. Our selection reflects these commonly studied areas.

For each experiment, we define a small but realistic grid around the original hyperparameter configurations, designed to be practical for replication. This allows us to assess the extent to which reported effects are sensitive to hyperparameter variation. We analyze which findings are unstable, in terms of changes in model rankings or in the sign and significance of statistical tests. Finally, we provide recommendations for experimental practice and reporting that improve the robustness of such findings.

This thesis discusses and potentially contributes to the field in the following ways.

1. We replicate three established papers in this field and find that part of their conclusions exhibit low stability under modest variations, suggesting that limited hyperparameter variation can already shift some linguistic outcomes.

2. Through discussions on the effect of hyperparameter tuning, types of fragile conclusions, and suggestions for stable conclusions, this paper contributes to the long-existing debate of reproducibility in the natural language processing field.

Chapter 2

Background

2.1 Language Modeling

Generative neural language models (LMs) are trained with the objective of next-token prediction, an idea that can be traced back to the origins of information theory, where the production of language sequences is formalized as a Markov process [72]. Early statistical and neural instantiations of this factorization include class-based n-gram models and neural probabilistic LMs [12, 7]. A sentence $\mathbf{x}$ is modeled as

$$P(\mathbf{x}) = \prod_{t=1}^{n} P(x_t \mid \mathbf{x}_{<t}) \tag{2.1}$$

whose form indicates that a language model assigns probability to a sequence by decomposing it into conditional predictions. Contemporary neural language models are trained to minimize cross-entropy loss under this tokenization.

A central debate is whether optimizing next-token prediction induces higher-level linguistic competence, or whether it mainly rewards surface-form heuristics [6]. While natural language is structured into six layers, phonetics, phonology, morphology, syntax, semantics, and pragmatics, which interact through layers, the next-token-prediction objective is doubtful whether it can model the higher-level structures of language. In cognitive modeling, this becomes an empirical question, tested by linking model surprisals to human behavioral and neural signals [75]. If models encode linguistically meaningful abstractions, their token probabilities should reflect sensitivity to linguistic features.

2.2 Linguistic Competence in Behavioral Experiments

Before the cognitive revolution from the 1950s to 1970s, the behaviorism trend has been dominating the psychology research area. The two research paradigms, behaviorism and cognitivism, diverge in the research method, where cognitivism carries out experiments to use conceptual or computational models to measure the human mental mechanisms in controlled settings, while behaviorism infers about the internal states of human minds by observing human behaviors [37, 77]. Unofficially borrowing these concepts from cognitive science, this paper classifies those which analyze on the model output and language model's last layer probability as behavioral experiments with language models, and those which probe the internal hidden states of language models as cognitivism experiments.

Researchers in machine learning interpretability designed a series of behavioral experiments to model language model's learning and understanding ability. Studies have examined phonological patterns, morphological productivity, syntactic competence, and semantic composition [1, 43, 57, *inter alia*]. Common approaches include targeted syntactic evaluations, psycho-linguistically inspired behavioral tests, or verification related to information theory rules [51, 33, 52, 86, 63, 4, *inter alia*]. These findings suggest a degree of alignment between model predictions and human processing, and typically hypothesize that training yields stable and well-formed linguistic representations.

2.3 Replication Crisis in Machine Learning

Reproducibility is a foundational requirement of empirical science. Since the early 2010s, concerns about a so-called *replication crisis*, a term coined by [68], have been raised across psychology, medicine, public health, and other domains, referring to systematic failures to reproduce published findings. These failures have been attributed to publication bias, underpowered studies, analytical flexibility, and selective reporting [45, 29]. [66] identified a series of this crises in domains like public health, psychology, medicine, etc.

Machine learning faces related but distinct challenges. Common threats to reproducibility include data leakage and contamination, benchmark overfitting, cherry-picked evaluations, insufficient reporting of experimental conditions, and undocumented preprocessing decisions [50]. Large-scale models further exacerbate the problem due to high computational cost, proprietary datasets, and incomplete release of training details.

In response, the community has introduced infrastructure and standards to improve transparency, including containerization tools (e.g., Docker), experiment tracking platforms, code

repositories linked to publications, and reporting checklists [58]. Nonetheless, systematic replication studies remain rare relative to the scale of published empirical claims.

2.4 Hyperparameter Optimization

Hyperparameter tuning strongly affects neural model behavior and observations from model training [48, 16, 22, 73, *inter alia*]. Small changes in learning rate, initialization, optimizer, etc., are crucial factors in determining whether the model will fall into a local minimum and whether high accuracy can ultimately be achieved [9, 2, 31, *inter alia*].

Recent empirical work has criticized drawing firm conclusions from models trained with arbitrary hyperparameters, emphasizing instead the need to examine how tuning choices affect performances [48, 16, 22]. [21] further argues that the process of hyperparameter optimization itself is worth systematic study.

Common practices of hyperparameter optimization include grid search, random search, and Bayesian search [59]. Grid search is time-consuming because it iterates over all combinations of training hyperparameters, whereas random search does not guarantee visiting all valid values of each hyperparameter. Bayesian optimization has become one of the first choices [76, 59] for hyperparameter tuning.

Bayesian optimization formalizes hyperparameter tuning as the global optimization of an unknown and expensive black-box objective function $f(\mathbf{x})$, where $\mathbf{x}$ denotes a hyperparameter configuration and f corresponds to a validation metric. Rather than exhaustively evaluating configurations, Bayesian optimization constructs a probabilistic surrogate model over f based on observed pairs $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ and sequentially selects new evaluation points by optimizing an acquisition function. A common choice of surrogate is a Gaussian process (GP), which defines a prior over functions, $f \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$, fully specified by a mean function m and a covariance kernel k [70]. Conditioning on observed data yields a posterior predictive distribution at a candidate point $\mathbf{x}_*$ with mean $\mu(\mathbf{x}_*)$ and variance $\sigma^2(\mathbf{x}_*)$, thereby quantifying both expected performance and epistemic uncertainty.

Chapter 3

Experiment A: Distributive Word Learning Pattern

This chapter examines whether the distributive word learning patterns reported by [14] can be robustly reproduced under controlled training conditions and modest hyperparameter variation.

The language-acquisition ability of neural networks, especially their word-acquisition patterns, is an indicator of their linguistic ability [63]. Discussions have been made on the divergence of models and humans in how they learn languages [42, 14, 28], and exclusive patterns that machines learn expressions [56, 20].

Learning theory has classified the acquisition of word-level patterns as distributional learning since it relies on the distributional statistics of input texts [49, 10].

3.1 Training Procedure

We first replicate the word acquisition experiment of [14], which compares neural models' word acquisition patterns with humans'. During training, the models' checkpoints are recorded at a specified frequency for further analysis, including fitting the learning curve, recording output token probabilities, and calculating age-of-acquisition indicators.

The training data is a mixture of BookCorpus [88] and WikiText-103 [26]. [1] The input sequences are paired and tokenized by a SentencePiece tokenizer [46].

[1] The exact versions of the two datasets we adopted are recorded in the https://huggingface.co/datasets/compling/word-acquisition

Due to the size of the training data, the training of this experiment takes days to finish. Besides, since [54] confirms the consistency of distributional signals over model architectures, we only use one architecture, the LSTM model, to replicate the experiment. For each set of hyperparameters, an LSTM model is trained on the dataset for 1M steps. All training runs take place on 4 A40 GPUs, with AdamW as the optimizer and Lambda LR scheduler. The checkpoint recording frequency is as described in Appendix B. The optimization space involved in this experiment includes the hyperparameters in Appendix A. We trained the LSTM on four configurations.

3.2 Claims to Be Examined

The conclusions of this paper can be summarized into the following ones: (A1) identified factors can predict words' age of acquisition (AoA) in language models, by the effects of five predictors (lexical class, word length, and concreteness), and language models exhibit a different learning pattern from children; (A2) language models acquire unigram and bigram statistics early in training.

For each word to be evaluated, its AoA is defined to be the midpoint of a sigmoid curve fitted over the word surprisals obtained on all saved checkpoints. A likelihood ratio test is conducted on five quantitative indicators, log-frequency, mean length of utterance (MLU), n-chars, concreteness, and lexical classes, to assess the significance of each predictor in predicting the AoA. The definition of each predictor is listed below.

- **Log frequency**: The log frequency is defined to be the natural logarithm of the word's per-100 token frequency.
- **MLU**: The mean length of utterance is defined to be the mean length of sequences containing a given word.
- **n-chars**: Defined as the number of letters in a word, n-chars is commonly considered as a proxy for the length of a word.
- **concreteness**: The concreteness is adopted from human-generated concreteness norms rated on a five-point scale.
- **lexical class**: The lexical class data is referred from the annotations in Wordbank [32]. The choices of lexical classes include Noun, Verb, Adjective, Function Words, and Other.

The sigmoid function is fitted through the SSlogis method in the R language. To determine the signs and significance of AoA predictors, a likelihood ratio test is performed comparing the full regression model with a reduced model that includes all predictors except the target variable. The signs of predictors are obtained from the general linear regression.

A set of Kullback-Leibler (KL) divergence curves between the model's word predictions and the unigram and bigram probabilities of the words is plotted to determine the acquisition trend of unigrams and bigrams. The definitions of the reference distributions used to calculate the KL divergence curves with the output logits are as follows.

- **Unigram**: For each token type, the unigram distribution is the raw token frequency in the training data.
- **Bigram**: For each token type, the bigram distribution is the token frequency dependent on the previous token in the training data.
- **One-hot**: The one-hot distribution indicates the true token distribution, and thus the KL-divergence between the model output and the one-hot distribution is the cross-entropy loss.
- **Uniform**: The uniform distribution is the random chance to predict a certain token type, regardless of their true distribution in the training data.

In the KL-divergence curves, the unigram probability is calculated by raw word frequency in the corpus, while the bigram probability is calculated by the conditional probability of a word depending on the previous word.

3.3 Results across Configurations

We replicated the main results of this paper, including the surprisal curve and the five predictors in predicting the AoA for the specific word 'eat'. Results are in Figure 3.1.

We sampled four sets of configurations on which we trained the models. Since the training data is extensive, the training time varies from 2 to 14 days per model on four A40 GPUs, depending on the number of layers and hidden dimension size. Due to time and resource limitations and because, after four sets of training, we found the learning patterns to be similar, we ceased training on additional configurations. These four configurations already cover the hyperparameters we designed and are thus representative enough.

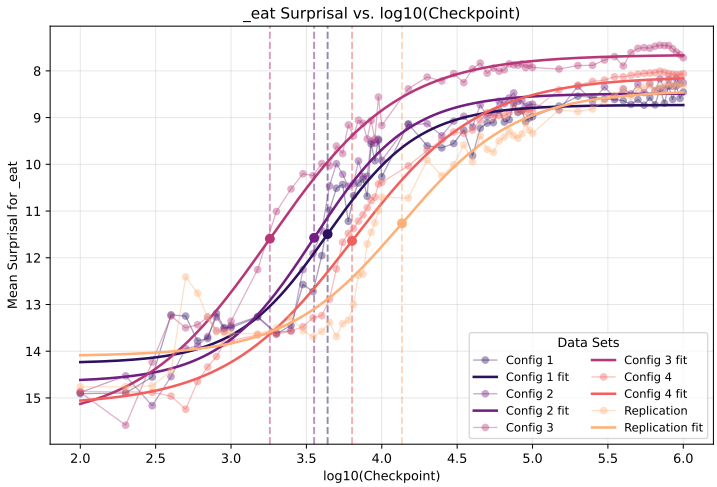
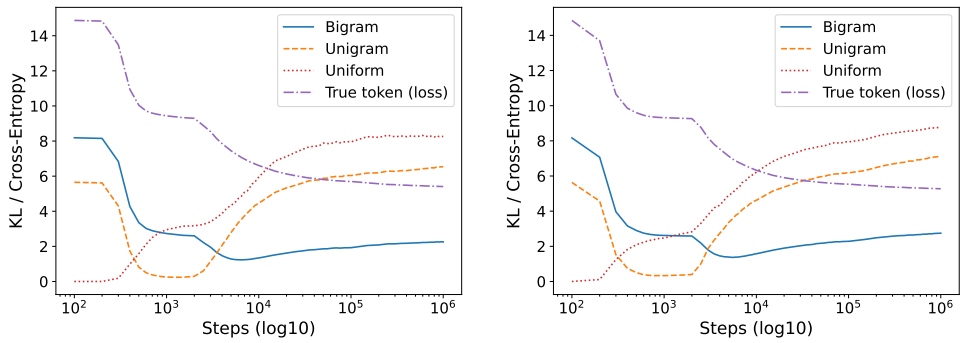


Figure 3.1: Learning curves of the word ‘eat’ for the LSTM model trained on each configuration.

For the case study word “eat,” these configurations produce sigmoid functions that are mostly similar, consistent with [14]’s characterization (A1) that neural models exhibit staged acquisition of lexicons. Hyperparameters primarily shift the position and the slope of the curves’ transitions. Examination of finer-grained conclusions about how predictors contribute to the AoA shows relatively robust findings as well, with effects varying only in significance, not in sign, as shown in Table 3.2.

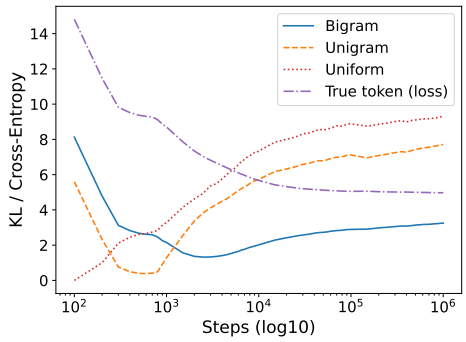
The fitted midpoints (Xmid) in Table 3.1 range by almost one log-step unit across configurations, while the scale parameter varies subtly as well, with some settings producing an earlier and steeper transition (e.g., configurations 2 and 3) and others a later transition. Thus, while the existence of a stable acquisition stage for a given word is robust, the precise inferred AoA is hyperparameter-dependent.

Regarding the claim about the unigram and bigram learning curves (A2), our hyperparameter variation also shows that the initially observed pattern holds across all our experiments, as illustrated in Figure 3.2. The KL-divergence curves between the model predictions and the unigram and bigram probabilities both increase after decreasing, matching the claim that the models always first overfit on unigrams and bigrams. As references, the curve for the uniform distribution keeps increasing, while that for the one-hot distribution, which equals the cross-entropy loss, remains decreasing.

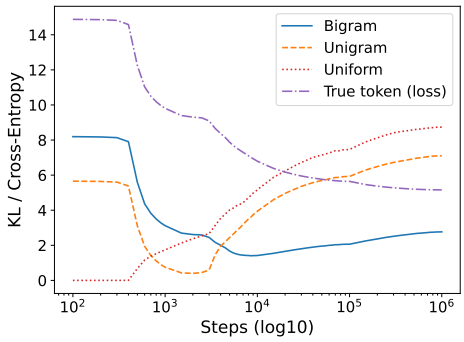


(a) Config 1.

(b) Config 2.



(c) Config 3.



(d) Config 4.

Figure 3.2: Curves of KL-divergence between the real distributions and learned predictions during training.

Config	Upper	Lower	Xmid	Scale
Replication	14.1073	8.4197	4.1340	0.3809
1	14.2582	8.7297	3.6408	0.3039
2	14.6603	8.4857	3.5512	0.3152
3	15.5302	7.6546	3.2575	0.4304
4	15.1542	8.1232	3.8027	0.4235

Table 3.1: In experiment A, parameters taken from fitted sigmoid curves on tested configurations. 'Upper' and 'lower' indicate the max and min y-values on the curve. 'Xmid' denotes the x-value of the midpoint of the curve. 'Scale' is a factor describing the range of the x-axis. The greater the 'scale' is, the flatter the curve will be.

3.4 Discussion

Observations from experiment A provide sufficient evidence on the stability of its conclusions, though a limited hyperparameter space is explored. These observations align with [54]'s finding that language models' behavior during word acquisition, which is a procedure highly dependent on the distributional statistics of training data, is consistent across model architectures.

Config	PPL	Log-freq	MLU	n-chars	Concreteness	Lexical class	R^2	HP
Base	54.80	*** (-)		*** (-)		***	0.93	
1	42.43	*** (-)		** (-)		***	0.88	Dropout -
2	38.75	*** (-)	* (+)	** (-)		***	0.88	#layers +
3	31.43	*** (-)	* (+)	*** (-)		*	0.88	#layers +, Batch size +
4	35.73	*** (-)	* (+)	** (-)		***	0.91	Layer size +, LR -
Children		*** (-)	*** (+)	** (+)	*** (-)	***	0.43	

Table 3.2: Coefficients in experiment A. Perplexity, significant predictors, and R^2 are reported in the original experiment and in the other four sets of configurations. Significant predictors are marked by asterisks ($p < 0.05$ *; $p < 0.01$ **; $p < 0.001$ ***). Signs of coefficients are notated in parentheses. The R^2 denotes the adjusted R^2 in a regression using all five predictors. The HP records the differences in hyperparameters between the corresponding setting and the base setting, with colors separating the model parameter and training parameter.

Chapter 4

Experiment B: Syntactic Generalization Ability

We then move to the syntactic generalization experiment, in which neural models are evaluated on their ability to judge unseen, ungrammatical syntactic tests. The general evaluation procedure is to train LMs on a natural language corpus, evaluate them on proposed syntactic test suites, and compare their performances. By plotting the relationship between test perplexities and models, we reveal differences in syntactic generalization ability across architectures.

Syntactic ability, in a general sense, is considered a cognitive ability [13]. The focus on language models' syntactic ability stems from the beginning of the cognitive revolution when Chomsky proposed the complexity hierarchy [18].

Recent research assessing the linguistic ability of neural language models has also increasingly focused on their syntactic skills, with the aim of understanding whether and how such models acquire human-like grammatical competence [39, 40, 34, 83, 30].

Formally, for a syntactic structure s instantiated in a sentence pair $(\mathbf{x}^+, \mathbf{x}^-)$, a model is sensitive to syntax if it assigns the surprisal $S(x)$ to each continuous range of tokens $\mathbf{t}$ meet the following criteria [34].

$$\prod_{i \in \mathbf{x}^+} S(t_i) < \prod_{i \in \mathbf{x}^-} S(t_i) \tag{4.1}$$

4.1 Training Setting

The corpus used for training is the BLLIP dataset [15], divided into four sizes: XS, SM, MD, and LG. For LSTM, the dataset is tokenized through a word tokenizer. The same UNK removal procedure is performed on the tokenized data as well. The vocab size after UNK removal matches the vocab size reported in the paper. For Transformer, the dataset is tokenized with the pre-trained GPT-2 tokenizer with a fixed vocab size. We verified that our word counts after data segmentation and unkification, and our token counts after tokenization, match the original paper.

Two models, LSTM and GPT-2, are trained with the Hugging Face Trainer [1], each on one H100 GPU. Both models use the AdamW optimizer and the linear scheduler. Each model has been trained three times on different random seeds for each configuration. Parameter space is documented in Appendix A.

4.2 Claims to Be Examined

Conclusions from this paper can be summarized into that, (B1) the results dissociate model perplexity and performance in the syntactic generalization task, suggesting that the two metrics represent complementary features of a language model's syntactic knowledge, and (B2) model architecture plays a more critical role than training data scale in yielding correct syntactic generalization.

In addition to computing the test-set perplexity after training on next-token-prediction tasks, each model checkpoint is evaluated on a syntactic test suite comprising 34 tests proposed by [41], which was later compiled into the SyntaxGym test suite [34]. The test suites are classified into the following types, each corresponding to a circuit that represents a certain ability to recognize syntax.

- **Agreement**: Agreement is a constraint on the feature values of two co-varying tokens. For example, English has subject-verb number agreement.
- **Center Embedding**: Center Embedding sentences are sentences recursively nested within each other. Subject and verbs must match in a first-in-last-out order, which means models must approximate a stack-like data structure in order to process these structures.

[1] The original paper trains GPT-2 on the Tensorflow framework, which is a difference.

- **Garden-Path Effect**: Garden-Path Effects are well-studied syntactic phenomena that result from tree-structural ambiguities that give rise to locally coherent but globally implausible syntactic parses.
- **Gross Syntax State**: Gross Syntactic Expectation is a processor's expectation for large syntactic chunks such as verb phrases or sentences, and is often set up by subordinating conjunctions such as while, although, and despite.
- **Licensing**: Licensing occurs when a particular token must exist within the scope of an upstream licenser token. Scope is determined by the tree-structural properties of the sentence.
- **Long-Distance Dependency**: Long-Distance Dependencies are covariations between two tokens that span a long distance in tree depth.

Each test judges whether a model can distinguish ungrammatical sentences by producing higher surprisals from grammatical ones. The accuracy on all tests defines the syntactic generalization (SG) score for that model checkpoint for further analysis. The SG scores and perplexities are averaged across three training runs with three random seeds for each training configuration.

4.3 Results across Configurations

Although the paper does not release the complete code or the training configuration, we seek to match the numbers and procedures described in the paper. We ensured the number of tokens and UNK tokens in data preprocessing matched those reported in the paper. We randomly selected five configurations from a commonly used parameter space. For each configuration, the model is trained three times with different random seeds until convergence on the validation set. All training runs yield validation-set perplexities similar to or smaller than those reported in the paper. Besides, a GPT-2 pretrained model [64] is loaded from Hugging Face's open-sourced weights[2] and tested on one L40S GPU, yielding similar SG scores (originally 0.7511, ours 0.7423). The evidence above confirms our replication is reliable.

Figure 4.1 shows the relationship between test perplexities and syntactic generalization (SG) scores across all hyperparameter configurations for LSTM and GPT-2. Same as [41], we observe that perplexities and SG scores are not monotonically related, since checkpoints with similar perplexities can exhibit markedly different SG scores (e.g., Configurations 1, 4, and 5 for LSTM

[2]https://huggingface.co/openai-community/gpt2

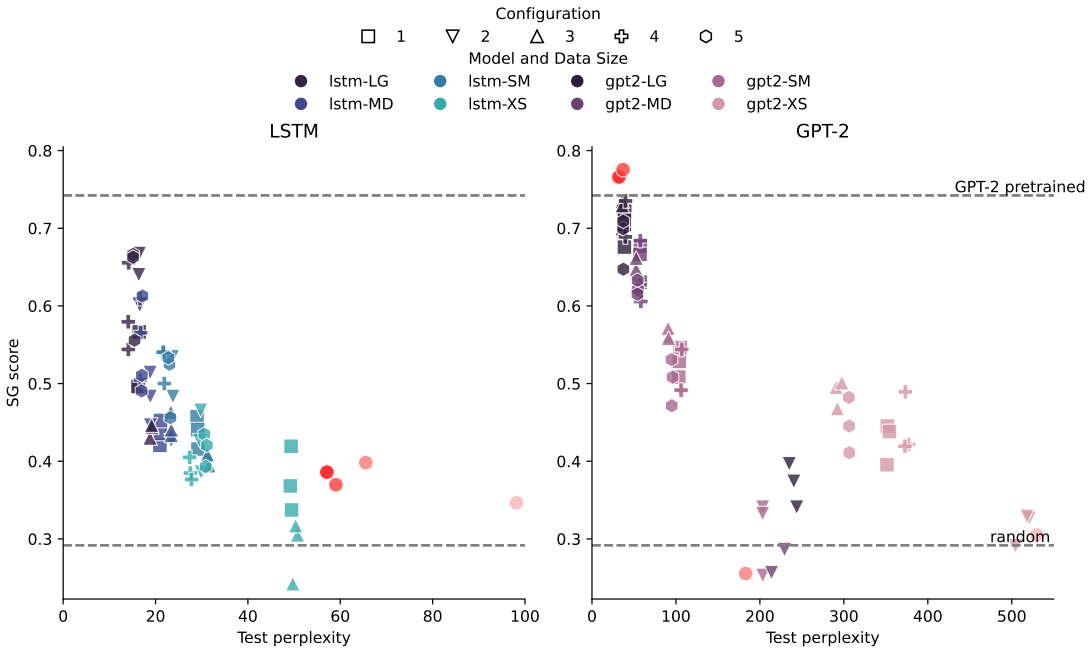


Figure 4.1: Comparison of the original paper's results, and the replicated test perplexities and SG scores for LSTM and GPT-2 on all configurations, each dot denoting one distinct training run. Red dots denote the original results. Same as [41], since GPT-2's losses have been converted from token-count-based into word-count-based, the perplexities of GPT-2 will increase in numbers, and thus perplexities are not directly comparable among model architectures.

LG, and Configurations 3, 4, and 5 for GPT-2 LG). This confirms the original paper's claim that perplexity and SG behave as complementary indicators of model quality rather than a single goodness metric, marking the stability of the coarse claim (B1).

In contrast, the conclusion B2 is not fully preserved from our results. For a fair comparison, we first rerun the mixed effect models on only the original paper's results of the two models we replicate. The mixed effect model is defined as Formula 4.2, where D denotes the number of words in training data, M the type of model architecture, S the class of test suite, and i each training instance.

$$\text{SG score} \sim ln(D) + M + (ln(D) + M \mid S) + (1 \mid i) \tag{4.2}$$

On the original results provided by [41], with LSTM and GPT-2 selected, both data size and model architecture still make statistically significant contributions in predicting the SG score.

The effect from the data size remain stronger than the model architecture (AIC=-371.67 when the data size predictor is ablated, and AIC=-374.25 when the model architecture is ablated).

After selecting only the configuration resulting in the best development set perplexities on each architecture, both data size and architecture make significant contribution to the SG scores. Predictor ablation experiment indicate that the data size predictor affects more to the model fit. (AIC=-476.51 when data size is ablated, while -497.61 when model architecture is ablated.)

Besides, the mixed effect regression 4.3 considering the effect from configurations show the configurations contribute a stronger effect to the model fit than the data size predictor (AIC=-2699.2 when data size is ablated, while AIC=-2695.6 when model architecture is ablated).

$$\text{SG score} \sim ln(D) + M + C + (ln(D) + M + C \mid s) + (1 \mid i) \tag{4.3}$$

4.4 Discussion

The results above show that experiment B is partially sensitive to hyperparameter variations. Across its conclusions, the coarse one claiming the specific linguistic ability is a distinct metric from the perplexity in next-token-prediction task still holds, echoing existing debate on the divergence of general model performance evaluation and specific linguistic abilities [4, 55]. In contrast, model ranking by linguistic ability can be easily affected by hyperparameter tuning, since well-chosen hyperparameters lead to better-trained models. Examining the model performances, it can be inferred that the differences in conclusions are likely due to the undertrained models in the original paper. Our LSTMs' perplexities are notably lower, and SG scores are higher than those reported in the original paper, possibly due to differences in the training framework, the GPU, or the hyperparameter choice, leading to under-tuned models.

4.5 Analysis on Effect of Tuning on SG Scores

Figure 4.2 illustrates the SG score changes for the five randomly chosen configurations and the Bayes tuning. Following the plots in [41], each point denotes the delta of the score on a specific test suite with the average score across all models, configurations, and test suites. Although the SG scores of models fluctuate on configurations yielding high perplexities, they stabilize as configurations with lower perplexities are found. Thus, the ranking of architectures in terms of syntactic ability becomes easier to decide.

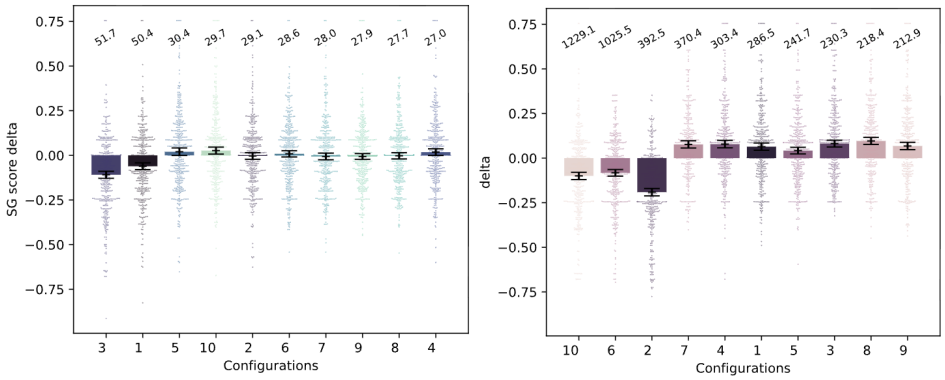


Figure 4.2: SG scores of each configuration, ranked by validation set perplexity on XS set. Each point represents the score delta relative to the average SG score across all models. Average validation-set perplexities are annotated over the bars. Error bars represent 95% confidence interval through bootstrapping.

4.6 Analysis on Syntax Ability by Test Suite

The syntactic bars per ability for each model is illustrated in Figure 4.3.

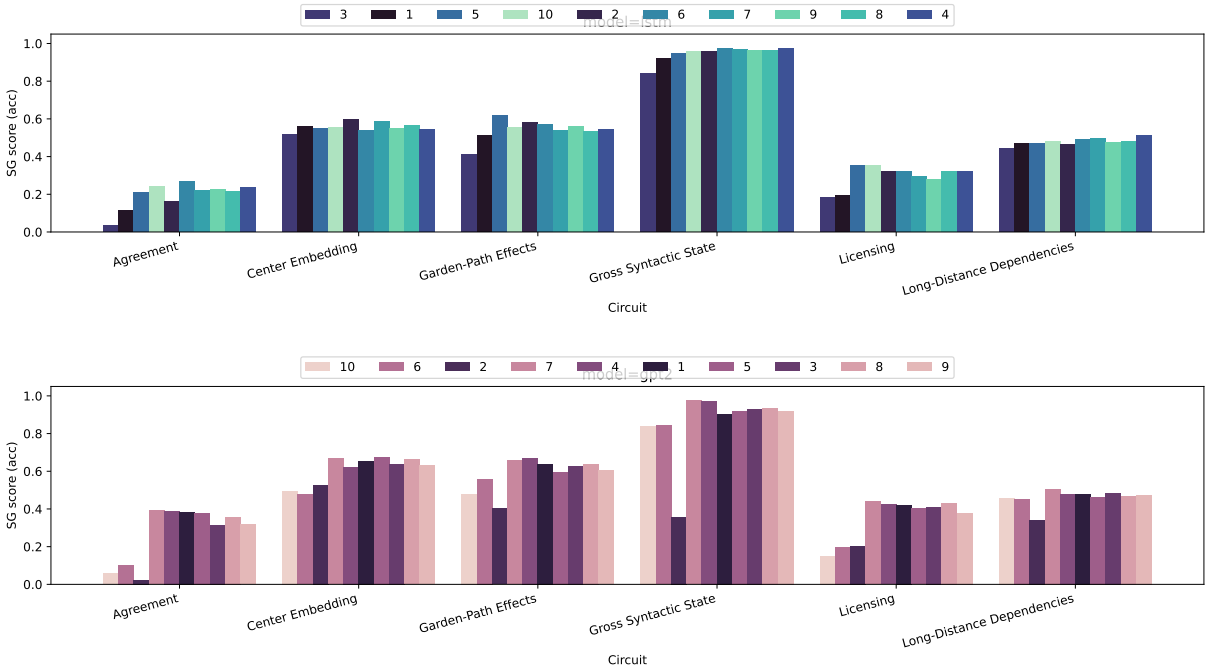


Figure 4.3: SG scores per syntactic ability of each configuration, ranked by validation set perplexity on XS set.

Chapter 5

Experiment C: Word Order Preference

Recent research claims that neural networks exhibit human-like word-order preferences. [82] and [47] demonstrated that cognitive-driven neural networks, such as those with memory limits, can learn languages whose word order occurs more frequently in human languages.

Human languages exhibit different orders of part-of-speech (POS) tagging [25]. Dominant word orders involve the order of subject (S), the predicate (V), and the object (O), and finer-grained orders involving the elements within these essential grammatical functions, such as the order of the adjective (Adj) and the noun (N), differ from language to language. The distribution of these typological orders is biased in number, e.g., SOV appears more frequently than SVO, and the order of certain POS taggings is considered to affect others, e.g., languages with VSO orders are more likely to have the adjectives after the nouns [38].

Such preferences of language models towards specific word orders are described as inductive bias by [82] and [47], and are tested to determine whether different model architectures exhibit word-order preferences that align with humans, as reflected in their empirical distributions in natural languages. Such preferences are often interpreted as a consequence of inductive bias, where architectural constraints restrict the hypothesis class (i.e., the set of functions a model can represent) prior to learning to prevent overfitting on training data [71].

5.1 Training Setting

This section replicates [47]'s experiment on verifying if memory-limited models have word-order preferences towards more frequent orders in natural languages. After training neural models with varying memory limits on 64 types of artificial languages with different word orders, the final test

perplexity for each model is documented. A correlation between the perplexity and the real-world frequency of each word order taken from WALS [25] is then calculated as the indicator of the similarity to human preferences.

The training data are artificial languages generated from one probabilistic context-free grammar (PCFG). This PCFG is designed to vary across six types of generation rules, each denoted as a switch (e.g., S $\rightarrow$ NP VP $[d_1 == 0]$ | VP NP $[d_1 == 1]$ is a rule yielding the values 'on' and 'off' for switch S). Thus, the grammar generates $2^6 = 64$ languages, each characterized by a distinct word-order. The identifier of the language is thus denoted as a switch vector $\mathbf{b} \in \{0, 1\}^6$, corresponding to the 7-digit grammar name with the 3rd digit marginalized. For each grammar, we sample 20,000 sentences to train each model. Each of such training runs is repeated 3 times to reduce the effect of randomness.

Three models, the simple recurrent neural network (SRN), long short-term memory (LSTM), and decoder-only Transformer (Transformer), are compared to verify this conclusion. The SRNs are assumed to be the most memory-limited, LSTMs in the middle, and Transformers are assumed to be the least memory-limited. On each grammar and each configuration, each model is trained on three separate but equivalent datasets. All models are trained with the AdamW optimizer and the Inverse-Sqrt scheduler. Each training run is run on a single L40 GPU. The hyperparameters involved in the tuning space from the *fairseq* training framework [60] are detailed in Appendix A. By varying a single hyperparameter each time, we selected seven configurations and trained on them.[1]

5.1.1 Claims to Be Examined

Conclusions of this experiment include (C1) typologically frequent word orders tend to have lower perplexity, and (C2) this relationship is better estimated by LMs with cognitively plausible biases, specific parsing strategies, and memory limitations. Due to computational constraints, we focus on the experiment related to memory limitations in C2, which assumes that the simple recurrent neural network (SRN), the most memory-limited model, has the most human-like word-order preference, LSTM is in the middle, and Transformer is the least.

5.2 Hypothesis and metrics.

The original experiment adopts Pearson's r correlation of the grammar's negative test perplexity obtained from training and the real-world data distribution in WALS, as the indicator of the

[1] Embedding size and hidden size are bound to change together.

Config	β_0	β_{SRN}	β_{Trans}	β_F	$\beta_{F,\text{SRN}}$	$\beta_{F,\text{Trans}}$	HP
Base	9.03 (***)	20.75 (***)	1.39 (*)				
Repl	6.35 (***)	13.93 (***)	0.29 (*)		-7.59 (*)		
1	6.47 (***)	16.78 (***)	0.23 (***)		-7.35 (*)		Batch size +
2	6.44 (***)	14.86 (***)					Dropout -
3	6.24 (***)	2.12 (***)	0.36 (***)				Early-stop +
4	6.47 (***)	16.78 (***)			-7.35 (*)		Layer size +
5	7.12 (***)	20.27 (***)	-0.58 (*)				#layer +
6	42.86 (***)	108.43 (***)	-32.42 (***)	-25.93 (***)	29.57 (**)		LR -
Best	6.24 (***)	0.29 (***)					

Table 5.1: Estimated coefficients related to the original paper's conclusion for all configurations in Experiment C. Coefficients with no significance are denoted as '-'. Significance codes: '***' for p < 0.001, '**' for p < 0.01, and '*' for p < 0.05. 'Repl' denotes the replication experiment. All adjusted R^2's are above 0.9. The HP records the differences in hyperparameters between the corresponding setting and the base setting, with colors separating the model parameter and training parameter.

model's human-like preference. A higher correlation is assumed to indicate that a model prefers frequent word-order patterns in natural languages. However, during replication, we found that most of the correlations reported in the paper or in our replication have high p-values, indicating insignificance (see Appendix C). Continuing to use Pearson's r thus yields no meaningful comparisons. We hence adopted a linear regression, defined as follows, to determine whether a model assigns lower PPLs to typologically frequent word orders.

$$\begin{aligned}\text{Test Perplexity} = \beta_0 \\ &+ \boldsymbol{\beta}_M^\top \mathbf{M} + \beta_F F \\ &+ \boldsymbol{\beta}_{FM}^\top (F \circ \mathbf{M}) \\ &+ \boldsymbol{\beta}_d^\top \mathbf{d} + \boldsymbol{\beta}_{dM} \top (\mathbf{d} \circ \mathbf{M}) \\ &+ \epsilon\end{aligned}$$

where $\mathbf{d}$ denotes the 6-digit feature vector (excluding the third digit), each as a 2-level categorical variable; F the frequency as a continuous value, $\mathbf{M}$ the model-class categorical indicator, and $\boldsymbol{\beta}$ the associated coefficients. The term $\mathbf{d} \circ \mathbf{M}$ denotes the element-wise interactions between the grammar-digit features and the model class. β_F and $\boldsymbol{\beta}_{FM}$ capture the frequencies' effect on the training perplexity, and how this effect varies across each model architecture. $\boldsymbol{\beta}_d$

and $\boldsymbol{\beta}_{dM}$ encode the effect of each specific grammar switch digit's effect on the test-set perplexity, and how well each model can make use of this digit. Thus, the paper's research questions can be translated to (C1) if more frequent word orders are preferred by language modes (β_F, $\boldsymbol{\beta}_{FM} < 0$), and (C2) if memory-more-limited models capture the natural word order distributions better than memory-less-limited models ($|\beta_{F,SRN}| > |\beta_F| > |\beta_{F,Trans}|$, where the vertical bars stand for absolute value). These paper-related coefficients $\boldsymbol{\beta}_M$, β_F, and $\boldsymbol{\beta}_{FM}$ will be focused on in the replication. Finer-grained analysis on the switch digit's effect is detailed in Appendix 5.5.

5.3 Results across Configurations

We first replicate the experiments by running [47]'s open-source training code on our machine. The result perplexities reached the same level to the paper, while the coefficients demonstrate similar sizes, signs, and significances to the original paper.

A comparison of the test-set perplexities between the original paper and the replicated runs is presented in Table 5.2. The sizes of the perplexities roughly match those in the original paper.

Experiment	SRN	LSTM	Transformer
Base	28.9020	8.8328	10.5339
Repl.	23.0733	7.7554	7.2007

Table 5.2: Test-set perplexities from replication of experiment C, comparing with results open-sourced by the original paper.

Across all configurations in Table 5.1, β_0 denotes the perplexity levels when LSTM is set as the base architecture level. β_{SRN} and β_{Trans} correspondingly show that switching the model architecture to SRN will generally increase the test-set perplexities, while for Transformer, the perplexities decrease. Meanwhile, out of four configurations where β_F and β_{FM} are significant, most of them are negative, indicating that increasing the frequency variable decreases test-set perplexity. This aligns with the original paper's claim (C1) that languages with more frequent word orders in WALS are easier for language models. However, given the lack of significance across the remaining configurations, the results from either the original paper or the replication run are considered to support this claim. Besides, since the significant negative terms appear only in configurations 1 and 4 for SRN and configuration 6 for LSTM (at the base architecture level), it is hard to argue that C1 is robust across hyperparameters.

The claim (C2) concerns architecture-specific word-order preferences, where memory-limited models exhibit more human-like preferences and correspond, in our linear regression, to the frequency-model interaction terms (β_{FM}). In our results, SRNs show stronger negative associations with perplexity across several configurations (1 and 4), whereas LSTMs and Transformers do not show significant effects in these settings. The expected order made by the assumption that LSTMs' negative effect should be greater than Transformers' is seldom presented, except in configuration 6. Configuration 6 also reverses the expected order between SRN and LSTM. Moreover, this expected order is not presented among the best configurations selected based on average validation-set perplexities over all languages (configuration 3 of LSTM, 3 of SRN, and 5 of Transformer), further undermining the original paper's conclusion C2.

5.4 Discussion

The instability of experiment C mainly results from the fact that its metrics involve model ranking and depend heavily on the Pearson correlation, a statistical test that requires checking the significance. However, the original paper's ignorance of p-values has led to the misreporting of insignificant results, which plausibly explains the high instability observed across hyperparameter settings.

5.5 Analysis on Word Order Preference by Switch

Table 5.3 presents the effect of a specific grammar switch digit on the perplexity $\boldsymbol{\beta}_d$ and its interaction with the model architecture class $\boldsymbol{\beta}_{dM}$. A reference to the meaning of each digit is presented in Table 5.4. In most cases with LSTM as the base architecture level, the effect of a single digit is not significant, except in configuration 7, where tuning the switch digit 1 on (VP NP) and digit 5 off (Adj N) decreases test-set perplexity. The architecture SRN has significant effects on test-set perplexities with the digits 2 (predicate-object order), 5 (adjective-noun order), 6 (attribute clause), and 7 (case agreement) in most cases. Switching the model architecture to Transformer does not present a significant effect within our hyperparameter space.

Config	$\beta_{\mathbf{d}}$	$\beta_{\mathbf{dM}}$	HP
Base		$d_6 \times SRN^{*}$(-), $d_7 \times SRN^{*}$(+)	
Repl		$d_2 \times SRN^{***}$(+), $d_5 \times SRN^{**}$(+), $d_6 \times SRN^{**}$(-), $d_7 \times SRN^{***}$(-)	
1		$d_2 \times SRN^{***}$(+), $d_6 \times SRN^{***}$(-), $d_7 \times SRN^{**}$(-)	Batch size +
2		$d_2 \times SRN^{***}$(+), $d_6 \times SRN^{***}$(-), $d_7 \times SRN^{*}$(-)	Dropout -
3			Early-stop +
4		$d_2 \times SRN^{***}$(+), $d_6 \times SRN^{***}$(-), $d_7 \times SRN^{**}$(-)	Layer size +
5		$d_2 \times SRN^{***}$(+), $d_7 \times SRN^{**}$(+)	#layer +
6	d_1^{*}(-), d_5^{**}(+)		
Best			LR -

Table 5.3: Estimated coefficients for all configurations on models' grammar-digit preferences in Experiment C. $\beta_{\mathbf{5\bullet}}$ summarizes the domain-model interactions ($d_1, \ldots, d_7$ * model). Coefficients with no significance are denoted as '-'. Significance codes: '***' for p < 0.001, '**' for p < 0.01, and '*' for p < 0.05. 'Repl' denotes the replication experiment. All adjusted R^2's are above 0.9. The HP records the differences in hyperparameters between the corresponding setting and the base setting, with colors separating the model parameter and training parameter.

Switch Digit	b = 0	b = 1	Additional Description
1	S → NP VP	S → VP NP	-
2	VP → NP V	VP → V NP	-
3	S_Comp → S Comp	S_Comp → Comp S	Object clause
4	NP* → NP Prep NP*	NP* → NP* Prep NP	-
5	NP → Adj N	NP → N Adj	-
6	NP → VP Rel N	NP → N Rel VP	Attribute clause
7	N_Subj/Obj → N Subj/Obj	N_Subj/Obj → Subj/Obj N	Case agreement

Table 5.4: Grammar definition of the artificial language adopted in the word order preference experiment.

Chapter 6

Discussion

6.1 Effects of Hyperparameters

Since experiments A and C are inappropriate for further analysis due to either stability or instability across hyperparameters, we chose a hyperparameter-sensitive experiment, experiment B, to further examine the variance in evaluation metrics related to linguistic findings as the hyperparameters change. Besides, analysis on experiment B would benefit from generalization since it trains the model on real-world datasets.

A Bayesian optimization [76, 59] is performed based on a Gaussian process[1], fit to the five existing training runs. The metric for determining the best model is average validation perplexity on the XS data size. In each optimization iteration, the configuration that maximizes the acquisition function, defined by the expected improvement (EI), is chosen as the next configuration to train on. The Bayes tuning continues for five iterations per model (denoted configurations 6 to 10). Model perplexities and SG scores on each configuration are detailed in 4.5.

A likelihood-ratio test is performed to assess the effect of hyperparameters on the models' syntactic ability by iteratively comparing a full model and a reduced model that preserves all other hyperparameters except the one being tested. Table 6.1 presents the signs and significance of each hyperparameter as a predictor of the SG scores of each model architecture. For both models, the number of layers (LSTM layers or Transformer blocks), dropout rate, and the initial learning rates are significant predictors of the model's SG score.

[1]The Gaussian process is built with Matérn kernel following the default of the *wandb* package [8].

Hyperparameter	Sign and Significance
Hidden Size	
Embedding Dimension	
Number of Layers	*** (+)
Dropout Rate	*** (-)
Initial Learning Rate	*** (+)

Hyperparameter	Sign and Significance
Hidden Size	
Number of Layers	*** (+)
Number of Heads	
Dropout Rate	*** (-)
Attention Dropout Rate	
Initial Learning Rate	*** (+)

Table 6.1: Hyperparameters as significant predictors of SG scores, in LSTM model (up) and GPT-2 model (bottom). Signs determined by the sign of coefficients in the overall regression, and significance by the likelihood ratio test.

6.2 Effects of Early Stopping

Research performing linguistic-driven experiments trains models with different numbers of steps, either with a fixed number of epochs [3, 14, 47, *inter alia*], or until the training converges on the validation set [82, 41, 84, *inter alia*]. However, whether a researcher can freely choose to train for a certain number of epochs or to train until convergence is under discussion.

This section further examines experiment B on the training dynamics, focusing on changes in their evaluation metrics. Based on the results from the Bayes tuning above, we compare their SG scores at the 1/4, 2/4, and 3/4 positions, as well as at the final checkpoints. The visualization is shown in Figure 6.1. The trajectories of the points demonstrate that as the training procedure continues, the SG scores for each data size generally increase, though in occasional cases, the scores at the final checkpoints are lower than those at the 3/4 checkpoints. The divergence across data sizes differs the most among these trajectories, as SG scores and perplexities diverge significantly at the early checkpoints but converge by the final checkpoints. Analysis on the divergence across runs can be affected by the choice of training epochs. Additionally, the SG scores for each epoch quantile are indicated by horizontal lines. Choosing to train for a certain number of epochs (e.g., 1/4) before convergence can mislead to a fake observation that LSTMs

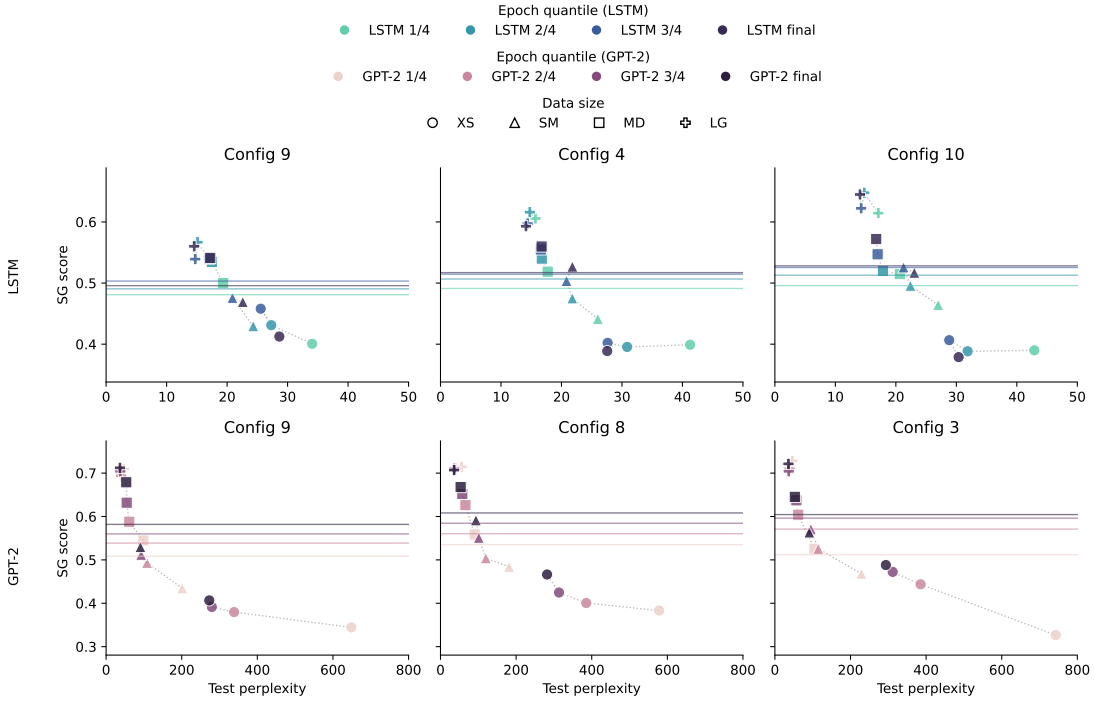


Figure 6.1: The dynamics of test-set perplexities SG scores according to training procedure. Early stopping is decided by convergence on validation-set, with best checkpoint loaded. LSTM Config 9 misses a 1/4 XS point since the corresponding training instances converge within three epochs.

and GPT-2's have similar SG scores.

6.3 Positions on Future Research

6.3.1 Hyperparameter Tuning

Hyperparameters can act as a substantial source of variability besides commonly accepted sources like random seeds or training data order. Empirical conclusions should be tuned over several rounds to produce better-trained models that better reflect the learning potential of specific model architectures. In addition, it is worth noting that the number of training epochs affects the model's behavior.

6.3.2 Fragile Conclusion

Our results reveal a gradient of sensitivity to hyperparameters across linguistic phenomena. Fine-grained lexical findings, such as staged acquisition and frequency effects, remain stable across a wide range of training configurations. Models' syntactic ability ranking shows moderate sensitivity. Typological word-order claims, which rely on subtle interactions among architectures, grammatical rules, and statistical tests, are the most sensitive to hyperparameter sweeps. This pattern suggests that claims relying on higher-order interactions require stronger robustness checks than claims based on lower-level distributional signals.

6.3.3 Reporting Standard

Experimental reports should document the hyperparameter range faithfully, tuning procedures, and other relevant sources of randomness (e.g., seeds, initialization, hardware), e.g., as the best practices [79] suggest for research in the natural language processing field. Architectural differences should be interpreted cautiously since they are not stable across hyperparameters. In addition to reporting best-performing checkpoints, researchers should report how conclusions vary under tuning. Additionally, statistical analyses should report their significance faithfully.

6.3.4 Replication Crisis

As a term originating in the early 2010s [69, 19], the replication crisis has led to substantial reforms in experimental design, reporting standards, and statistical methodology across psychology and related sciences.

Concerns about reproducibility have increasingly extended to experimental linguistics and machine learning. In natural language processing and machine learning fields, empirical findings are often sensitive to implementation details, hyperparameter choices, random seeds, and data preprocessing decisions [24, 22, 5]. Reported gains may disappear under alternative seeds or modest retuning, raising questions about the stability of conclusions.

A common research pattern in ML-driven linguistics involves: (i) proposing a novel architecture or modeling idea, demonstrated via limited experiments as proof-of-concept; (ii) follow-up studies investigating robustness, ablations, or generalization across datasets and settings.

Type-(i) work often operates under computational constraints and may under-explore hyperparameter space or variance across runs. Type-(ii) work, while sometimes perceived as incremental, is essential for assessing generality and effect stability. Empirical comparisons without adequate

variance reporting risk over-interpreting architectural differences that are in fact hyperparameter-dependent [53].

Statistical evaluation practices further complicate the picture. The choice of statistical test, evaluation metric, or multiple-comparison correction directly shapes reported conclusions [24]. Repeated experimentation without pre-specified stopping criteria or correction procedures creates opportunities for implicit multiple testing, p-hacking, and selective reporting [35, 74].

6.3.5 Exploratory and confirmatory data analysis

Exploratory data analysis (EDA) [78] and confirmatory data analysis (CDA) [80] provide a principled distinction between hypothesis generation and hypothesis testing.

Exploratory hypothesis testing (EHT) involves unplanned, post-hoc analyses aimed at discovering patterns, whereas confirmatory hypothesis testing (CHT) involves pre-specified hypotheses and analysis plans [67]. Winter [85] argues that exploratory work should transparently document analytic flexibility, while confirmatory work should pre-register procedures and adhere to pre-defined statistical criteria.

In ML and experimental linguistics, this distinction is often blurred: hyperparameter tuning, model selection, and dataset filtering frequently occur prior to final statistical testing, yet are rarely treated as part of the inferential procedure. Because model selection itself constitutes data-dependent hypothesis refinement, failing to separate exploration from confirmation inflates Type I error and weakens evidential claims.

Future work should explicitly distinguish exploratory and confirmatory phases. Exploratory analyses should emphasize robustness checks, variance reporting, and effect size estimation. Confirmatory studies should predefine evaluation protocols, seeds, model selection criteria, and statistical tests. Such separation does not constrain innovation; it clarifies which conclusions are provisional and which are inferentially warranted.

Chapter 7

Conclusion

This thesis introduces the concept of hyperparameter stability in linguistically driven experiments with language models by replicating three representative research studies and examining the variance of their conclusions across several training hyperparameter settings. We find that the results from the experiments above are, to some extent, all affected by the hyperparameter choice, among which some observations alter or reverse the original conclusion. From an experimental perspective, linguistically driven experiments that observe models' distributional learning signals, such as word-level learning curves, tend to be stable. Rankings of model abilities and significance of statistical tests are more sensitive to hyperparameter choices, which are common metrics for higher-level signals such as syntactic ability or language-typological preferences. Conclusions drawn from partially trained models are particularly sensitive to variation induced by hyperparameter choices. To improve the reproducibility of future experiments, we argue that research from this domain should view hyperparameter choice as a source of randomness equally important as other sources, such as random seed and graphic processing unit type, and that conclusions should be based on adequately trained models or robustness checks across settings.

References

[1] Morris Alper and Hadar Averbuch-Elor. Kiki or bouba? sound symbolism in vision-and-language models. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[2] Christian Arnold, Luka Biedebach, Andreas Küpfer, and Marcel Neunhoeffer. The role of hyperparameters in machine learning models and how to tune them. *Political Science Research and Methods*, 12(4):841–848, 2024.

[3] Christoph Aurnhammer and Stefan Frank. Comparing gated and simple recurrent neural network architectures as models of human sentence processing, 2018.

[4] Lisa Beinborn and Nora Hollenstein. *Cognitive plausibility in natural language processing*. Springer, 2023.

[5] Anya Belz, Shubham Agarwal, Anastasia Shimorina, and Ehud Reiter. A systematic review of reproducibility research in natural language processing. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 381–393, Online, 2021. Association for Computational Linguistics.

[6] Emily M. Bender and Alexander Koller. Climbing towards NLU: On meaning, form, and understanding in the age of data. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5185–5198, Online, 2020. Association for Computational Linguistics.

[7] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in*

Neural Information Processing Systems 13, Papers from Neural Information Processing Systems 2000, Denver, CO, USA, pages 932–938. MIT Press, 2000.

[8] Lukas Biewald. Experiment tracking with weights and biases, 2020. Software available from wandb.com.

[9] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, 2006.

[10] Gemma Boleda. Distributional semantics and linguistic theory. *Annual Review of Linguistics*, 6(1):213–234, 2020.

[11] Xavier Bouthillier, César Laurent, and Pascal Vincent. Unreproducible research is reproducible. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 725–734. PMLR, 2019.

[12] Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jenifer C. Lai, and Robert L. Mercer. Class-based *n*-gram models of natural language. *Computational Linguistics*, 18(4):467–480, 1992.

[13] Andrew Carnie. *Syntax: A generative introduction*. John Wiley & Sons, 1969.

[14] Tyler A. Chang and Benjamin K. Bergen. Word acquisition in neural language models. *Transactions of the Association for Computational Linguistics*, 10:1–16, 2022.

[15] Eugene Charniak, Don Blaheta, Niyu Ge, Keith Hall, John Hale, and Mark Johnson. Bllip 1987-89 wsj corpus release 1. *(No Title)*, 2000.

[16] Dami Choi, Christopher J. Shallue, Zachary Nado, Jaehoon Lee, Chris J. Maddison, and George E. Dahl. On empirical comparisons of optimizers for deep learning. *ArXiv preprint*, abs/1910.05446, 2019.

[17] Noam Chomsky. Syntactic structures, 1957.

[18] Noam Chomsky. *Aspects of the Theory of Syntax*. Number 11. MIT press, 1967.

[19] Open Science Collaboration. Estimating the reproducibility of psychological science. *Science*, 349(6251):aac4716, 2015.

[20] Ionut Constantinescu, Tiago Pimentel, Ryan Cotterell, and Alex Warstadt. Investigating Critical Period Effects in Language Acquisition through Neural Language Models, 2024.

[21] A. Feder Cooper, Yucheng Lu, Jessica Forde, and Christopher De Sa. Hyperparameter optimization is deceiving us, and how to stop it. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 3081–3095, 2021.

[22] Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. Show your work: Improved reporting of experimental results. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2185–2194, Hong Kong, China, 2019. Association for Computational Linguistics.

[23] Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. Show your work: Improved reporting of experimental results. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2185–2194, Hong Kong, China, 2019. Association for Computational Linguistics.

[24] Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. The hitchhiker's guide to testing statistical significance in natural language processing. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia, 2018. Association for Computational Linguistics.

[25] Matthew S. Dryer and Martin Haspelmath, editors. *WALS Online (v2020.4)*. Zenodo, 2013.

[26] Stephen Merity et al. WikiText-103. *ArXiv preprint*, abs/1609.07843, 2016.

[27] Allyson Ettinger. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48, 2020.

[28] Linnea Evanson, Yair Lakretz, and Jean Rémi King. Language acquisition: do children and language models follow similar learning stages? In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics:*

ACL 2023, pages 12205–12218, Toronto, Canada, 2023. Association for Computational Linguistics.

[29] Dustin A Fife and Joseph Lee Rodgers. Understanding the exploratory/confirmatory data analysis continuum: Moving beyond the "replication crisis". *American Psychologist*, 77(3):453, 2022.

[30] Matthew Finlayson, Aaron Mueller, Sebastian Gehrmann, Stuart Shieber, Tal Linzen, and Yonatan Belinkov. Causal analysis of syntactic agreement mechanisms in neural language models. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *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)*, pages 1828–1843, Online, 2021. Association for Computational Linguistics.

[31] Luca Franceschi, Michele Donini, Valerio Perrone, Aaron Klein, Cédric Archambeau, Matthias Seeger, Massimiliano Pontil, and Paolo Frasconi. Hyperparameter Optimization in Machine Learning, 2024.

[32] Michael C Frank, Mika Braginsky, Daniel Yurovsky, and Virginia A Marchman. Wordbank: An open repository for developmental vocabulary data. *Journal of child language*, 44(3):677–694, 2017.

[33] Richard Futrell, Ethan Wilcox, Takashi Morita, Peng Qian, Miguel Ballesteros, and Roger Levy. Neural language models as psycholinguistic subjects: Representations of syntactic state. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *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)*, pages 32–42, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.

[34] Jon Gauthier, Jennifer Hu, Ethan Wilcox, Peng Qian, and Roger Levy. SyntaxGym: An online platform for targeted evaluation of language models. In Asli Celikyilmaz and Tsung-Hsien Wen, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 70–76, Online, 2020. Association for Computational Linguistics.

[35] Andrew Gelman and Eric Loken. The garden of forking paths : Why multiple comparisons can be a problem , even when there is no " fishing expedition " or " p-hacking " and the research hypothesis was posited ahead of time. 2019.

[36] Steven N. Goodman, Daniele Fanelli, and John PA Ioannidis. What does research reproducibility mean? *Science translational medicine*, 8(341):341ps12–341ps12, 2016.

[37] George Graham. Behaviorism. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Spring 2023 edition, 2023.

[38] Joseph H Greenberg et al. Some universals of grammar with particular reference to the order of meaningful elements. *Universals of language*, 2:73–113, 1963.

[39] Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. Colorless green recurrent networks dream hierarchically. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1195–1205, New Orleans, Louisiana, 2018. Association for Computational Linguistics.

[40] Jennifer Hu, Sherry Yong Chen, and Roger Levy. A closer look at the performance of neural language models on reflexive anaphor licensing. In Allyson Ettinger, Gaja Jarosz, and Joe Pater, editors, *Proceedings of the Society for Computation in Linguistics 2020*, pages 323–333, New York, New York, 2020. Association for Computational Linguistics.

[41] Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger Levy. A systematic assessment of syntactic generalization in neural language models. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1725–1744, Online, 2020. Association for Computational Linguistics.

[42] Philip A. Huebner, Elior Sulem, Fisher Cynthia, and Dan Roth. BabyBERTa: Learning more grammar with small-scale child-directed language. In Arianna Bisazza and Omri Abend, editors, *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 624–646, Online, 2021. Association for Computational Linguistics.

[43] Mete Ismayilzada, Defne Circi, Jonne Sälevä, Hale Sirin, Abdullatif Köksal, Bhuwan Dhingra, Antoine Bosselut, Duygu Ataman, and Lonneke Van Der Plas. Evaluating morphological compositional generalization in large language models. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1270–1305. Association for Computational Linguistics, 2025.

[44] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.

[45] Max Korbmacher, Flavio Azevedo, Charlotte R Pennington, Helena Hartmann, Madeleine Pownall, Kathleen Schmidt, Mahmoud Elsherif, Nate Breznau, Olly Robertson, Tamara Kalandadze, et al. The replication crisis has led to positive structural, procedural, and community changes. *Communications Psychology*, 1(1):3, 2023.

[46] Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Eduardo Blanco and Wei Lu, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, 2018. Association for Computational Linguistics.

[47] Tatsuki Kuribayashi, Ryo Ueda, Ryo Yoshida, Yohei Oseki, Ted Briscoe, and Timothy Baldwin. Emergent Word Order Universals from Cognitively-Motivated Language Models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14522–14543. Association for Computational Linguistics, 2024.

[48] Hugo Larochelle, Dumitru Erhan, Aaron C. Courville, James Bergstra, and Yoshua Bengio. An empirical evaluation of deep architectures on problems with many factors of variation. In Zoubin Ghahramani, editor, *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007*, volume 227 of *ACM International Conference Proceeding Series*, pages 473–480. ACM, 2007.

[49] Alessandro Lenci. Distributional models of word meaning. *Annual review of Linguistics*, 4(1):151–171, 2018.

[50] Igor L. Markov. Reevaluating google's reinforcement learning for ic macro placement. *Commun. ACM*, 67(11):60–71, 2024.

[51] Rebecca Marvin and Tal Linzen. Targeted syntactic evaluation of language models. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Brussels, Belgium, 2018. Association for Computational Linguistics.

[52] Clara Meister, Tiago Pimentel, Patrick Haller, Lena Jäger, Ryan Cotterell, and Roger Levy. Revisiting the Uniform Information Density hypothesis. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 963–980, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

[53] Gábor Melis, Chris Dyer, and Phil Blunsom. On the state of the art of evaluation in neural language models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[54] James A. Michaelov, Roger P. Levy, and Benjamin K. Bergen. Language model behavioral phases are consistent across architecture, training data, and scale, 2025.

[55] Raphaël Millière. Language models as models of language, 2024.

[56] Kanishka Misra and Kyle Mahowald. Language Models Learn Rare Phenomena from Less Rare Phenomena: The Case of the Missing AANNs. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 913–929. Association for Computational Linguistics, 2024.

[57] Jeff Mitchell and Mirella Lapata. Composition in distributional models of semantics. *Cognitive science*, 34(8):1388–1429, 2010.

[58] David Moreau, Kristina Wiebels, and Carl Boettiger. Containers for computational reproducibility. *Nature Reviews Methods Primers*, 3(1):50, 2023.

[59] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[60] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. In Waleed Ammar, Annie Louis, and Nasrin Mostafazadeh, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.

[61] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[62] Fernando C Pereira. Formal grammar and information theory: together again? *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 358:1239 – 1253, 2000.

[63] Eva Portelance and Masoud Jasbi. The Roles of Neural Networks in Language Acquisition. *Language and Linguistics Compass*, 18(6):e70001, 2024.

[64] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[65] Terry Regier. The emergence of words: Attentional learning in form and meaning. *Cognitive Science*, 29(6):819–865, 2005.

[66] Felipe Romero. Philosophy of science and the replicability crisis. *Philosophy Compass*, 14(11):e12633, 2019.

[67] Mark Rubin and Chris Donkin. Exploratory hypothesis tests can be more compelling than confirmatory hypothesis tests. *Philosophical Psychology*, 37(8):2019–2047, 2024.

[68] Frank L Schmidt and In-Sue Oh. The crisis of confidence in research findings in psychology: Is lack of replication the real problem? or is it something else? *Archives of scientific psychology*, 4(1):32, 2016.

[69] Stefan Schmidt. Shall we really do it again? the powerful concept of replication is neglected in the social sciences. *Review of general psychology*, 13(2):90–100, 2009.

[70] Matthias Seeger. Gaussian processes for machine learning. *International journal of neural systems*, 14(02):69–106, 2004.

[71] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[72] Claude E Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

[73] Haoyue Shi, Karen Livescu, and Kevin Gimpel. On the role of supervision in unsupervised constituency parsing. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7611–7621, Online, 2020. Association for Computational Linguistics.

[74] Joseph P Simmons, Leif D Nelson, and Uri Simonsohn. False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological science*, 22(11):1359–1366, 2011.

[75] Nathaniel J. Smith and Roger Levy. The effect of word predictability on reading time is logarithmic. *Cognition*, 128(3):302–319, 2013.

[76] Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical bayesian optimization of machine learning algorithms. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 2960–2968, 2012.

[77] Paul Thagard. Cognitive Science. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Winter 2023 edition, 2023.

[78] John Wilder Tukey et al. *Exploratory data analysis*, volume 2. Springer, 1977.

[79] Dennis Ulmer, Elisa Bassignana, Max Müller-Eberstein, Daniel Varab, Mike Zhang, Rob van der Goot, Christian Hardmeier, and Barbara Plank. Experimental standards for deep learning in natural language processing research. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2673–2692, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics.

[80] Eric-Jan Wagenmakers, Ruud Wetzels, Denny Borsboom, Han LJ van der Maas, and Rogier A Kievit. An agenda for purely confirmatory research. *Perspectives on psychological science*, 7(6):632–638, 2012.

[81] Alex Warstadt and Samuel R Bowman. What artificial neural networks can tell us about human language acquisition. In Shalom Lappin and Jean-Philippe Bernardy, editors, *Algebraic Structures in Natural Language*, pages 17–60. CRC Press, 2022.

[82] Jennifer C. White and Ryan Cotterell. Examining the inductive bias of neural language models with artificial languages. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *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)*, pages 454–463, Online, 2021. Association for Computational Linguistics.

[83] Ethan Wilcox, Pranali Vani, and Roger Levy. A targeted assessment of incremental processing in neural language models and humans. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *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)*, pages 939–952, Online, 2021. Association for Computational Linguistics.

[84] Ethan Gotlieb Wilcox, Jon Gauthier, Jennifer Hu, Peng Qian, and Roger Levy. On the Predictive Power of Neural Language Models for Human Real-Time Comprehension Behavior, 2020.

[85] Bodo Winter. *Statistics for Linguists: An Introduction Using R*. Routledge, 2019.

[86] Tianyang Xu, Tatsuki Kuribayashi, Yohei Oseki, Ryan Cotterell, and Alex Warstadt. Can Language Models Learn Typologically Implausible Languages?, 2025.

[87] Changbing Yang, Franklin Ma, Freda Shi, and Jian Zhu. LingGym: How Far Are LLMs from Thinking Like Field Linguists?, 2025.

[88] Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 19–27. IEEE Computer Society, 2015.

APPENDICES

Appendix A

Tuning Space

Random sampling space. The hyperparameter configurations randomly sampled for three experiments are recorded in Table A.1.

Bayes tuning space. The Bayes tuning space of experiment B is documented in Table A.2.

Config	batch size	learning rate	number layers	embedding size	dropout
Replication	64	5e-4	3	768	0.1
1	64	5e-4	3	768	0.0
2	64	5e-4	3	1,536	0.3
3	256	5e-4	4	768	0.1
4	256	5e-5	4	1,536	0.3

(a) Configurations for experiment A, the word acquisition experiment.

Config	embedding size	hidden size	num layers	num heads	dropout	attention dropout	learning rate
LSTM 1	128	512	4	-	0.1	-	5e-5
LSTM 2	64	256	2	-	0.0	-	1e-3
LSTM 3	256	256	2	-	0.1	-	5e-5
LSTM 4	128	512	2	-	0.1	-	1e-3
LSTM 5	256	256	4	-	0.1	-	1e-3
GPT-2 1	768	3,072	12	12	0.0	0.0	5e-5
GPT-2 2	384	1,536	4	4	0.3	0.0	1e-3
GPT-2 3	768	3,072	12	12	0.1	0.1	5e-5
GPT-2 4	384	1,536	12	12	0.1	0.1	5e-5
GPT-2 5	768	3,072	6	6	0.1	0.1	5e-5

(b) Configuration for experiment B, the syntax test experiment. The symbol'-' denotes not applicable hyperparameters to a model. Configurations 6 to 10 are selected through Bayes tuning.

Config	batch size	dropout	epoch	hidden size	number layers	learning rate
Replication	256	0.3	10	512/64	2	5e-4
1	512	0.3	10	512/64	2	5e-4
2	256	0.3	10	512/64	2	5e-4
3	256	0.1	10	512/64	2	5e-4
4	256	0.3	5000	512/64	2	5e-4
5	256	0.3	10	1024/128	2	5e-4
6	256	0.3	10	512/64	4	5e-4
7	256	0.3	10	512/64	2	5e-5

(c) Configurations for experiment C, the word order preference experiment. Since this experiment focuses on cross-model-architecture comparison, each configuration is shared across models.

Table A.1: Configurations of all experiments performed.

Model	Hyperparameter	Space	Type
LSTM	Embedding size	{64, 128, 256}	Discrete
	Hidden size	{256, 512}	Discrete
	Numebr of layers	{2, 3, 4}	Discrete
	Dropout rate	{0.0, 0.1, 0.2, 0.3}	Discrete
	Initial learning rate	(1e-5, 1e-3)	Continuous variable, log-uniformly sampled
GPT-2	Embedding size	{384, 512, 768, 1024}	Discrete
	Number of layers	{6, 8, 12}	Discrete
	Number of heads	{6, 8, 12}	Discrete
	Dropout rate	{0.0, 0.1, 0.2, 0.3}	Discrete
	Attention Dropout	{0.0, 0.1, 0.2}	Discrete
	Initial learning rate	(1e-6, 1e-4)	Continuous variable, log-uniformly sampled

Table A.2: Hyperparameter optimization space of experiment B during Bayes tuning.

Appendix B

Implementation Difference in Each Experiment

Data deduplication in [47]. In [47]'s data generation procedure, which is a direct reusing the open-source PCFG and code from [82]'s paper, around six hundred sentences out of 100,000 sentences were overlapping between the training set and the test set during data generation. We regenerated the same number of sentences after adding a duplication removal procedure.

Checkpointing frequency in [14]. Due to the storage limit, we slightly reduced the checkpointing frequencies to the following ones.

- Checkpointing at every 100 steps during the first 1,000 steps.
- Checkpointing at every 500 steps during the first 10,000 steps.
- Checkpointing at every 5,000 steps during the first 100,000 steps.
- Checkpointing at every 50,000 steps during the first 1,000,000 steps.

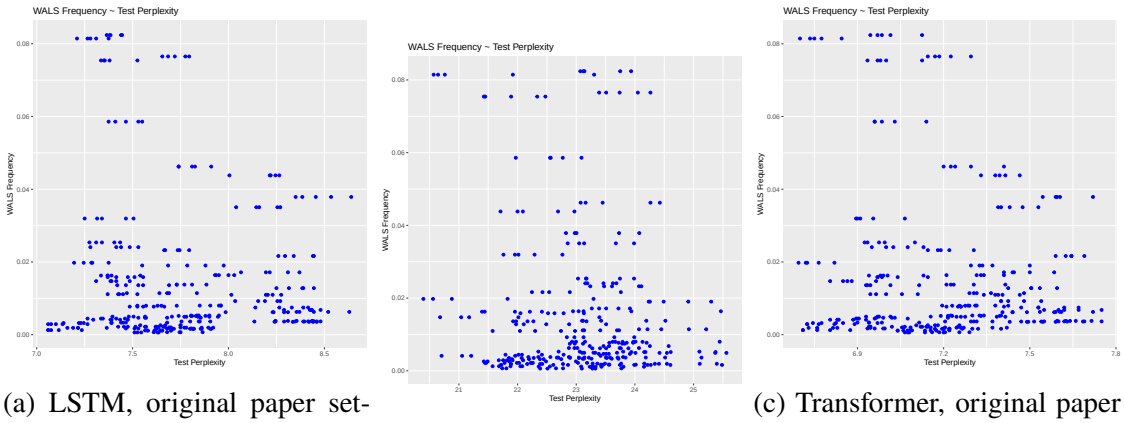
Appendix C

P-value in [47] and change of the metric.

[47] originally adopts Pearson's r between the grammars' frequencies and test perplexities on models as the indicator of the models' human-like word order preferences. Upon checking the original data, we found that many correlation tests yielded p-values > 0.05, indicating statistical insignificance. Our replication with the same methods also yields high p-values. Scatter plots in Figure C.1 provide further evidence of the weak correlation. Thus, we would regard Pearson's r an insufficiently reliable metric in modeling this research question.

However, the insignificant correlation between two variables does not cancel the possibility of their interaction in predicting another variable. Since the research question concerns about which model architecture captures human-like frequency the most in producing the perplexities, it is more natural to instead employ the linear regression model discussed in the paper body, with model architecture and frequency as two predictors of the test-set perplexity,

Comparison of the original results and replicated results under this regression metric is illustrated in Figure C.2.



(a) LSTM, original paper setting

(b) SRN, original paper setting

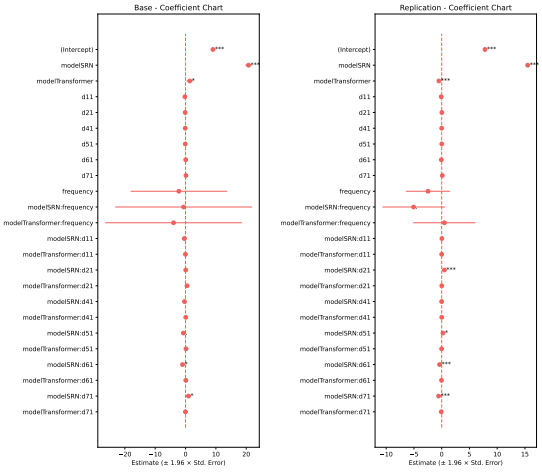
(c) Transformer, original paper setting

(d) LSTM, tuned best setting

(e) SRN, tuned best setting

(f) Transformer, tuned best setting

Figure C.1: An example of poor correlation with high p-values in Pearson correlation testing: WALS frequency and model perplexity of grammars in Experiment C. The low correlation and high probability to reject the correlation demonstrated by these scatter plots hinder both drawing a conclusion and observing a variance resulting from hyperparameters.



(a) Original paper. (b) Replication run.

Figure C.2: In experiment C, the word order preference experiment, coefficients from fitted linear regression on the training results in the original paper and replication run.