How Do Neural Sequence Models Generalize?
Local and Global Context Cues for Out-of-Distribution Prediction

Anthony Bau and **Jacob Andreas**
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
{abau, jda}@mit.edu

Abstract

After a neural sequence model encounters an unexpected token, can its behavior be predicted? We show that RNN and transformer language models exhibit structured, consistent generalization in out-of-distribution contexts. We begin by introducing two idealized models of generalization in next-word prediction: a *local context model* in which generalization is consistent with the last word observed, and a *global context model* in which generalization is consistent with the global structure of the input. In experiments in English, Finnish, Mandarin, and random regular languages, we demonstrate that neural language models *interpolate* between these two forms of generalization: their predictions are well-approximated by a log-linear combination of local and global predictive distributions. We then show that, in some languages, *noise* mediates the two forms of generalization: noise applied to input tokens encourages global generalization, while noise in history representations encourages local generalization. Finally, we offer a preliminary theoretical explanation of these results by proving that the observed interpolation behavior is expected in log-linear models with a particular feature correlation structure. These results help explain the effectiveness of two popular regularization schemes and show that aspects of sequence model generalization can be understood and controlled.

1 Introduction

Neural language models (LMs) play a key role in language processing systems for tasks as diverse as machine translation, dialogue, and automated speech recognition (Baziotis et al., 2020; Sordoni et al., 2015; Mikolov et al., 2010). These LMs, which model distributions over words in context via recurrent, convolutional, or attentional neural networks, have been found to consistently outperform finite-state approaches to language modeling based on hidden Markov models (Kuhn et al., 1994) or n-gram statistics (Miller and Selfridge, 1950). But improved predictive power comes at the cost of increased model complexity and a loss of transparency. While it is possible to characterize (and even control) how finite-state models will behave in previously unseen contexts, generalization in neural LMs is not nearly as well understood.

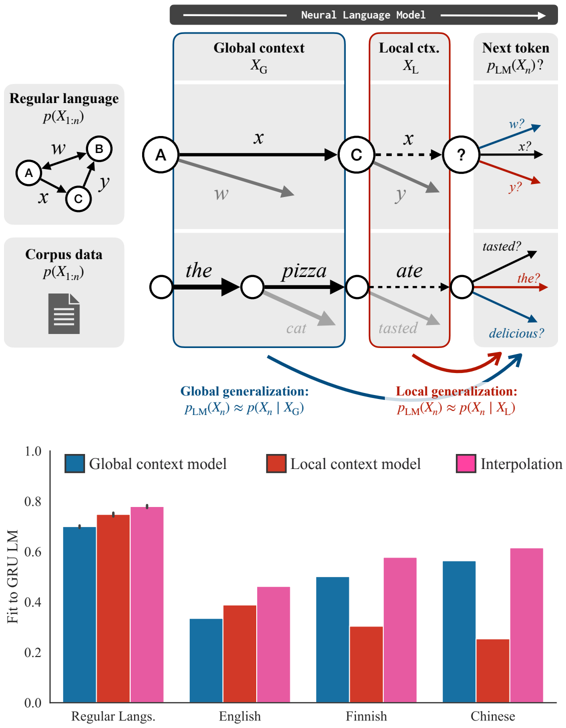

Figure 1: We develop formal models of the predictions of neural language models in *surprising contexts* in which local information (e.g. the most recent token) and global information (e.g. the rest of the sentence) conflict (top). In these out-of-distribution contexts, predictors trained on both synthetic and natural languages favor either local or global information, but are best approximated by an *interpolation* of a local-only and global-only predictor (bottom).

Consider the following sentence prefixes:

(a) *The pandemic won't end children can...*
(b) *Let him easter...*
(c) *After we ate the pizza, the pizza ate...*

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 5513–5526
November 7–11, 2021. ©2021 Association for Computational Linguistics

Each of these prefixes should be assigned a low probability under any reasonable statistical model of English: (a) is missing a word, (b) has a noun used in place of a verb, and (c) a features a selectional restriction violation.[1] When exposed to these surprising contexts, what word will language models predict next? For finite-state models of language, the answer is clear: n-gram models back off to the shortest context in which statistics can be reliably estimated (e.g. just the final word; Katz 1987), and hidden Markov models explicitly integrate the possibility of an unexpected part-of-speech transition and an unexpected word choice (Freitag and McCallum, 1999). But in neural models, model behavior in-distribution provides little insight into behavior in novel contexts like the ones shown in (a–c).

Characterizing neural LMs' behavior on inputs like these is important for many reasons—including evaluating their robustness, characterizing their effectiveness as models of human language processing, and identifying inductive biases relevant to deployment in new tasks. This paper offers three steps toward such a characterization:

1. We present an empirical description of neural LM behavior in out-of-distribution contexts like the ones shown in (a–c). We introduce two idealized models of prediction in these contexts: a *local context model* in which generalization is consistent with the last word observed (ignoring global sentence structure), and a *global context model*, in which generalization is consistent with the global structure of the input (ignoring unexpected words). In experiments on English, Finnish, Mandarin, and a collection of random regular languages, we show that neural LM behavior is reasonably well approximated by either the local or global context model, and even better predicted by an *interpolation* of the two: neural LMs reconcile conflicting information from local and global context by modeling their contributions independently and combining their predictions post-hoc (Fig. 1).

2. We further show that, in regular languages, *noise* introduced at training time modulates the relative strength of local and global context in this interpolation: input noise (in the form of random word substitution) encourages global generalization, while history noise (dropout applied to recurrent states or self-attention layers) encourages local generalization. These effects are small, but point toward a potential role for noise-based regularization schemes in controlling out-of-distribution behavior.

3. Finally, we offer a preliminary mathematical explanation of the observed results by demonstrating that this interpolation behavior arises in any regularized log-linear model with separate local and global context features that are individually predictive of future tokens.

Despite the complexity of current neural LMs, these results show that aspects of their out-of-distribution generalization can be characterized, controlled, and understood theoretically.

2 Background

Generalization in count-based LMs Before the widespread use of neural approaches in NLP, statistical approaches to language modeling were typically *defined* by explicit independence assumptions governing their generalization in contexts never observed in the training data. For example, n-gram models (Miller and Selfridge, 1950; Shannon, 1951) ignore global sentence structure in favor of a local context of at most n words. By contrast, latent-variable language models based on finite-state machines (Kuhn et al., 1994) (or more expressive automata; Chelba and Jelinek 1998, Pauls and Klein 2012) explicitly incorporate information from the long-range context by conditioning next-word prediction on abstract global states constrained by global sentence structure. In models of both kinds, behavior in contexts unlike any seen at training time is be explicitly specified via *backoff* and *smoothing* schemes aimed at providing robust estimates of the frequency of rare events (Good, 1953; Katz, 1987; Kneser and Ney, 1995). Like past work on backoff and smoothing, our work in this paper attempts to provide a general mechanism for both prediction and control in more complex, black-box neural LMs.

Generalization in feature-based and neural LMs Such mechanisms are necessary because, with the advent of feature-rich approaches to

[1] In fact, all three are examples of naturally occurring text: (a) was originally published (as a typo) on the New York Times homepage (@pennstate452, 2021), (b) appears in Hopkins (1918) and (c) in by Bench (2013).

language modeling—including log-linear models (Rosenfeld, 1996) and neural network models (Bengio et al., 2003; Mikolov et al., 2010; Vaswani et al., 2017)—the kinds of structured, engineered generalization available in finite-state models of language have largely been lost. Current models clearly generalize to new linguistic contexts (including those with semantic content very different from anything seen at training time; Radford et al. 2019). But the precise nature and limits of that generalization—especially its robustness to unusual syntax and its ability to incorporate information about global sentence structure—remain a topic of ongoing study.

Current work largely focuses on controlled, linguistically motivated tests of generalization: measuring models' ability to capture long-range agreement, movement, and licensing phenomena on diagnostic datasets (Gauthier et al., 2020). For example, Linzen et al. (2016) show that while RNNs are capable of storing the information necessary to enforce subject–verb agreement, the language modeling training objective does not encourage it; McCoy et al. (2020) demonstrate that RNN models for a question formation task favor linear generalizations over hierarchical ones (roughly, lexical generalizations over syntactic ones) on out-of-distribution inputs. Rather than focusing on a specific language or class of linguistic phenomenon, our work in this paper aims to provide a general-purpose framework for reasoning about generalization in neural sequence models across contexts and languages.

Generalization beyond NLP The generalizations investigated in paper involve instances of covariate shift—a change in the distribution $p(x)$ for a conditional model $p(y \mid x)$—which has been extensively investigated in more general machine learning settings (e.g. Storkey, 2009). Outside of NLP, there have been several attempts to describe more abstract inductive biases native to RNNs and transformers, including work focused on compositionality (Liška et al., 2018; Lake and Baroni, 2018; Weber et al., 2018) and even more generic algorithmic priors (Lan et al., 2021; Kharitonov and Chaabouni, 2020). Here we focus on the architectures and context shifts relevant to language processing tasks. We validate our models of generalization using real models trained on natural data and explain them in terms of measurable properties of these data distributions.

3 Models of Generalization

Consider the example contexts shown in (a–c). Each is an extremely unlikely sentence prefix, featuring text that is globally inconsistent with English syntax or semantic constraints. In such contexts, is it possible to predict *a priori* what a neural LM trained on language data will do next?

We can formalize the situation depicted in these examples as follows: Let $p(X_{1:n}) = p(X_1, X_2, \ldots, X_n)$ be a distribution over sentences with tokens X_i, and let

$$p_{\mathrm{LM}}(X_{1:n}) = \prod_{i=1}^{n} p_{\mathrm{LM}}(X_i \mid X_{1:i-1}) \qquad (1)$$

be a learned approximation to this distribution produced by an autoregressive model of the conditional distribution $p_{\mathrm{LM}}(X_n \mid X_{1:n-1})$. We will consider each **context** $X_{1:n-1}$ to comprise a **global context** $X_{\mathrm{G}} = X_{1:n-2}$ (all but the last word) and a **local context** $X_{\mathrm{L}} = X_{n-1}$ (the last word in the context). Then, given some thresholds ϵ and τ, we will call a context $(X_{\mathrm{G}}, X_{\mathrm{L}})$ **surprising** if for some thresholds ϵ and τ,

$$p_{\mathrm{LM}}(X_{\mathrm{L}} | X_{\mathrm{G}}) < \epsilon \qquad (2)$$

(the juxtaposition of X_{G} and X_{L} is low-probability), while:

$$p_{\mathrm{LM}}(X_{\mathrm{L}}) > \tau \qquad (3)$$

and each for each i

$$p_{\mathrm{LM}}(X_{\mathrm{G},i} \mid X_{\mathrm{G},1:i}) > \tau \qquad (4)$$

(X_{G} and X_{L} are high-probability marginally). In example (c), $X_{\mathrm{G}} =$ *the pizza*, $X_{\mathrm{L}} =$ *ate*. Given a language model p_{LM}, we wish to understand whether $p_{\mathrm{LM}}(X \mid X_{\mathrm{G}}, X_{\mathrm{L}})$ has systematic or predictable structure in surprising contexts—can it be explained in terms of statistics of the underlying distribution p or the behavior of p_{LM} in unsurprising contexts?

In the remainder of this section, we describe a set of candidate hypotheses about what this next-token distribution might look like, and in Section 4 evaluate the extent to which these hypotheses accurately predict the true behavior of p_{LM}.

3.1 Local and global models of generalization

We focus on two idealized models of the generalization that might be exhibited by neural language models.

Local context model In this model, we hypothesize that predictors reconcile the conflicting information from X_{G} and X_{L} by ignoring the global component of the context, and making the next-token distribution *locally* consistent with the last token seen, regardless of global sentence structure. We denote this model of generalization $\tilde{p}_{\mathrm{L}}$:

$$\tilde{p}_{\mathrm{L}}(X_n \mid X_{\mathrm{G}}, X_{\mathrm{L}}) \triangleq p(X_n \mid X_{\mathrm{L}}) \,. \tag{5}$$

$\tilde{p}_{\mathrm{L}}$ implements a form of backoff common in n-gram language models: faced with a long context in which the data distribution is unknown, models discard long-range information and use higher-quality estimates from a shorter context. We previously defined $X_{\mathrm{L}} = X_{n-1}$, so experiments with $\tilde{p}_{\mathrm{L}}$ will predict that neural LMs behave like bigram models; this could be naturally generalized to local contexts consisting of more than a single word.

$\tilde{p}_{\mathrm{L}}$ can also be viewed as the hypothesis that NLMs implement a particular kind of lossy-context model (Futrell and Levy, 2017; Futrell et al., 2020), who note that "local contextual information plays a privileged role in [human] language comprehension"; as we will see, this appears to be the case for some neural models as well. Sequence models with backoff may also be given a hierarchical Bayesian interpretation (Teh, 2006).

Global context model As an alternative, we consider the possibility that predictors rely *exclusively* on the global component of the context, ignoring the unexpected final token:

$$\begin{aligned}\tilde{p}_{\mathrm{G}}(X_n \mid X_{\mathrm{G}}, X_{\mathrm{L}}) &\triangleq p(X_n \mid X_{\mathrm{G}}) \\ &= \sum_v p(X_n \mid X_{n-1} = v, X_{\mathrm{G}}) \\ &\quad \times p(X_{n-1} = v \mid X_{\mathrm{G}}) \,. \end{aligned}\tag{6}$$

In the language of count-based models, this amounts to the hypothesis that NLMs generalize as *skip-gram* models (Goodman, 2001; Guthrie et al., 2006), performing a kind of reverse backoff to context prior to the most recent word. In the global context model, it is the most recent word, and not the rest of the context, that as treated as a possible source of noise to be marginalized out rather than conditioned on.

3.2 Interpolated models

Even when combined in surprising ways, both the local and global context are likely to carry useful information about the identity of the next word. Indeed, models and features implementing both kinds of context representation have been found useful in past work on language modeling (Goodman, 2001). It is thus natural to consider the possibility that neural LMs *interpolate* between the local context and global context models, combining evidence from $p(X_n \mid X_{\mathrm{L}})$ and $p(X_n \mid X_{\mathrm{G}})$ when there is no evidence for the specific context $p(X_n \mid X_{\mathrm{L}}, X_{\mathrm{G}})$.

We consider two ways in which this evidence might be combined:

Linear interpolation In this model,

$$\tilde{p}_{+}^{\lambda} \triangleq \lambda \cdot \tilde{p}_{\mathrm{L}} + (1-\lambda) \cdot \tilde{p}_{\mathrm{G}} \,. \tag{7}$$

Here we predict generalization according to a direct weighted combination of $\tilde{p}_{\mathrm{L}}$ and $\tilde{p}_{\mathrm{G}}$, with the relative importance of the two hypotheses controlled by a parameter $\lambda \in [0,1]$. Informally, this hypothesis assigns non-negligible probability to next tokens that are consistent with either base hypothesis. Similar interpolation schemes were proposed for n-gram modeling by Jelinek and Mercer (1980).

Log-linear interpolation In this model,

$$\tilde{p}_{\times}^{\lambda_1,\lambda_2} \overset{\triangle}{\propto} \tilde{p}_{\mathrm{G}}^{\lambda_1} \cdot \tilde{p}_{\mathrm{L}}^{\lambda_2} \,. \tag{8}$$

That is

$$\log \tilde{p}_{\times}^{\lambda_1,\lambda_2} = \lambda_1 \log \tilde{p}_{\mathrm{G}} + \lambda_2 \log \tilde{p}_{\mathrm{L}} - \log Z \tag{9}$$

for λ_1 and $\lambda_2 \in [0,1]$ and some contextual normalizing constant Z that depends on $X_{1:n-1}$. Here, probabilities from the two base hypotheses are added in log-space then renormalized; informally, this has the effect of assigning non-negligible probability to next tokens that are consistent with *both* base hypotheses. A similar approach was proposed for count-based language modeling by Klakow (1998).

4 Experiments

Which of these models (if any) best describes the empirical behavior of neural LMs trained on real datasets? In this section, we present two sets of evaluations. The first aims to characterize how well $\tilde{p}_{\mathrm{L}}$, $\tilde{p}_{\mathrm{G}}$, and combinations of the two predict the out-of-distribution behavior of RNN (Elman, 1990) and transformer (Vaswani et al., 2017) language models with standard training. The second explores whether these training procedures can be modified to control the relative strength of local and global generalization.

Both sets of experiments investigate the behavior of RNN and transformer LMs on a diverse set of datasets: first, a collection of *random regular languages* in which the true data distribution $p(X)$ can be precisely modeled; second, a collection of *natural language* datasets from three languages (Mandarin Chinese, English, and Finnish) which vary in the flexibility of their word order and the complexity of their morphology. We begin with a more detailed discussion of models and datasets in Section 4.1; then describe generalization experiments in Section 4.2 and control experiments in Section 4.3.

4.1 Preliminaries

Data: Formal languages The first collection of evaluation datasets consists of a family of *random regular languages*. We begin by generating three deterministic finite automata, each with 8 states and a vocabulary of 128 symbols. Using the algorithm in Appendix C, we randomly add edges to the DFA to satisfy the following constraints: (1) every state is connected to approximately 4 other states, and (2) each symbol appears on approximately 4 edges. States are marked as accepting with probability $\frac{1}{2}$.

Experiments on these carefully controlled synthetic languages are appealing for a number of reasons. First, because we have access to the true generative process underlying the training data, we can construct arbitrarily large training sets and surprising evaluation contexts $(X_{\mathrm{G}}, X_{\mathrm{L}})$ that are guaranteed to have *zero* probability under the training distribution, ensuring that our experiments cleanly isolate out-of-distribution behavior. Second, the specific construction given above means that "evidence" for the local and global models of generalization is balanced: no tokens induce especially high uncertainty over the distribution of states that can follow, and no states induce especially high uncertainty over the set of tokens they can emit, meaning that a preference for local or global generalization must arise from the model rather than the underlying data distribution.

In experiments on these datasets, we generate *training* examples via a random walk through the DFA, choosing an out edge (or, if available, termination) uniformly at random from those available in the current state. We generate *surprising test* examples by again sampling a random walk, then appending a symbol that *cannot* be produced along any out-edge from that random walk's final state. We compute $\tilde{p}_{\mathrm{L}}$ and $\tilde{p}_{\mathrm{G}}$ using the ground-truth distribution from each DFA.

In regular languages, the local context model thus hypothesizes that *lexical* information governs out-of-distribution prediction, predicting that LM outputs are determined by the set of states attached to an edge labeled with the surprising symbol. Conversely, the global context model hypothesizes that *structural* information governs out-of-distribution prediction: LM outputs are determined by the set of states reachable from the last state visited before the surprising symbol.

RNN experiments use gated recurrent units (Cho et al., 2014) with a word embeddings of size 128 and a single hidden layer of size 256. Transformer experiments use a hidden size of 256, 4 self-attention layers, and ReLU nonlinearities. Both models are trained with the Adam optimizer and a learning rate of 3e−4 on 128,000 examples.

Data: Natural languages The second collection of evaluation datasets use natural language data. We conduct experiments on English, Finnish, and Mandarin Chinese. These languages exhibit varying degrees of morphological complexity and freedom of word order, with Finnish at one extreme (morphologically complex and freely ordered) and Mandarin at the other.

English data comes from the WMT News Crawl corpus (Barrault et al., 2019). We used a 20,000-sentence subset of sentences from articles from 2007, tokenized using the SentencePiece byte-pair encoding (Kudo and Richardson, 2018) with a vocabulary size of 2^{14}. We used a 2,000-sentence held-out set for validation. Finnish data comes from the Turku Dependency Treebank (Haverinen et al., 2014), and Chinese data from the Simplified GSD Treebank, both included in the Universal Dependencies corpus (Nivre et al., 2020). These datasets are already tokenized; for the Chinese data we used the existing tokenization, limited by a vocabulary size of $2^{14} - 2$ with added "unknown" and "end-of-sentence" tokens. For Finnish we also used the SentencePiece byte-pair encoding with a vocabulary size of 2^{14}.

To generate surprising natural language sentences $(X_{\mathrm{G}}, X_{\mathrm{L}})$, we first select X_{G} by truncating sentences from the validation set to uniformly random lengths. We then run our best trained model p_{LM} to determine $p_{\mathrm{LM}}(X_{n-1}|X_{\mathrm{G}})$, and

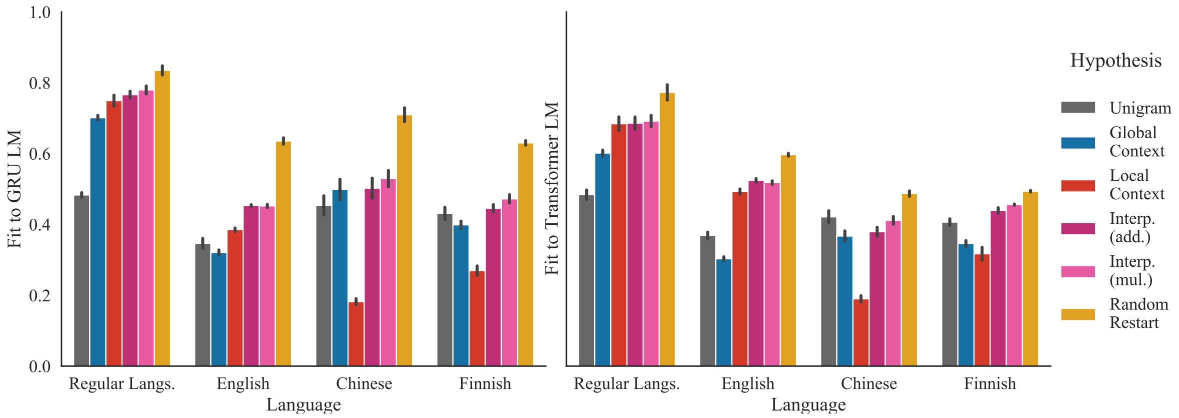

Figure 2: Accuracy $\mathrm{acc}(\tilde{p}, p_{\mathrm{LM}})$ of predicted generalization for various hypotheses $\tilde{p}$, with black lines showing one standard deviation across 5 (GRU) or 4 (transformer) random restarts. In some cases, generalization hypotheses are nearly as predictive as new neural models trained on the same data, suggesting that they explain most of extrapolation behavior that can be derived from data alone. Multiplicative interpolation is consistently a bit better than additive interpolation. Which of the two base hypotheses performs best (global or local generalization) varies substantially across languages.

choose a token X_{L} uniformly from among the set $\{X : p_{\mathrm{LM}}(X|X_{\mathrm{G}}) < \frac{1}{198}, X \in L\}$ where L is the set of the 198 most-common tokens by unigram count (200 less the "unknown" and "end-of-sentence" tokens used in Chinese). In the framework of Equations 2–4, ϵ is set to $\frac{1}{198}$ and τ to the smallest probability assigned in context to an in-distribution token.

To compute generalization model predictions on natural language data, we estimate $\tilde{p}_{\mathrm{L}}$ from bigram counts in the training set: $\tilde{p}_{\mathrm{L}}(X_n|X_L) = \frac{\mathrm{count}(X_n, X_L)}{\mathrm{count}(X_L)}$. To estimate $\tilde{p}_{\mathrm{G}}$, we train a second, random restart of the model $p_{\mathrm{LM}'}$. We then estimate $\tilde{p}_{\mathrm{G}}$ using one step of beam search in $p_{\mathrm{LM}'}$ with a beam of size 15:

$$\tilde{p}_{\mathrm{G}}(X_n|X_{\mathrm{G}}) \approx \sum_{i=1}^{15} p_{\mathrm{LM}'}(v_i|X_{\mathrm{G}}) p_{\mathrm{LM}'}(X_n|X_{\mathrm{G}}, v_i) \quad (10)$$

where the v_i range over the 15 top predicted tokens after X_{G}. Given a trained model that performs well on the *in-distribution* validation set, we will have $p_{\mathrm{LM}}(v|X_{\mathrm{G}}) \approx p(v|X_{\mathrm{G}})$ and $p_{\mathrm{LM}}(X_n|X_G, v) \approx p(X_n|X_G, v)$, and therefore that Eq. (10) gives a good approximation of $\tilde{p}_{\mathrm{G}}$.[2]

For each natural language datasets, we trained GRUs with 2 hidden layers and word embedding and hidden sizes of 1024, and transformers with 4 heads, 2 layers, and hidden sizes of 512. All models were optimized with Adam using a learning rate of 3e-4 on shuffled length-aligned batches of up to 128 for 15 epochs. The model with the best held-out performance was then selected.

4.2 Which model of generalization fits best?

Given a dataset of surprising contexts $\{(X_{\mathrm{G}}, X_{\mathrm{L}})_i\}$, a hypothesis $\tilde{p}$, and a trained model p_{LM}, we compute the accuracy of the hypothesis $\tilde{p}$ as

$$\mathrm{acc}(\tilde{p}, p_{\mathrm{LM}}) = 1 - \mathrm{err}(\tilde{p}, p_{\mathrm{LM}}) \quad (11)$$

where

$$\mathrm{err}(\tilde{p}, p_{\mathrm{LM}}) = \frac{1}{n} \sum_{X_{\mathrm{G}}, X_{\mathrm{L}}} \delta(p_{\mathrm{LM}}(\cdot \mid X_{\mathrm{G}}, X_{\mathrm{L}}), \tilde{p}(\cdot \mid X_{\mathrm{G}}, X_{\mathrm{L}}))$$

and δ is the total variation distance

$$\delta(p_1, p_2) = \frac{1}{2} \|p_1 - p_2\|_1 . \quad (12)$$

In other words, we measure the accuracy of each hypothesis by computing the average ℓ_1 distance between the hypothesized and true probability histograms across surprising contexts. $\mathrm{acc}(\tilde{p}, p_{\mathrm{LM}})$ is between 0 and 1; a large value indicates that $\tilde{p}$ is a good approximation to p_{LM}.[3]

[2] We compute this quantity with a second language model in order to prevent information about the true model's out-of-distribution behavior from leaking into our prediction.

[3] The use of a bounded distance measure is important—intuitively, we should regard a hypothesis as accurate even if it makes very inaccurate predictions in a small fraction of contexts. However, the specific choice of measure does not appear to be very important; defining err in terms of the Jensen–Shannon divergence gives similar results.

For hypotheses $\tilde{p}$, we use (1) the local and global context (Section 3.1), and (2) *optimal* linear and log-linear interpolations between them (Section 3.2), choosing settings for λ that minimize error on the evaluation set itself. To provide context for these results, we report the accuracy of a **unigram baseline**, which predicts the unigram distribution $p(X_n)$ independent of the context. We additionally report the error obtained by a **random restart**—a new model $p_{\theta'}$ trained from scratch (with a different initialization) on the same data as p_{LM}; this model provides a rough upper bound on how much of p_{LM}'s prediction can be explained by structural properties of the data distribution itself. err was computed from 100 samples for regular languages and 200 samples for natural languages.

Results are shown in Fig. 2. In each language, either the local or global model is a good fit to observed generalization behavior. There are substantial differences across languages: the local context model is a good predictor for regular languages and English, but a poor predictor for Finnish and Chinese, suggesting that generalization behavior is data-dependent but not tied to word order or morphological complexity. In general, GRU generalization is more predictable than transformer generalization. Finally, interpolation often substantially outperforms either base hypothesis (Fig. 3), with log-linear interpolation slightly more predictive than linear interpolation. In several cases, interpolated hypotheses approach the accuracy of a randomly retrained predictor, suggesting that they capture much of the generalization behavior that is determined by the underlying data alone.

4.3 What controls interpolation?

The previous section showed that $\tilde{p}_{\times}^{\lambda_1,\lambda_2}$ gives the best fit to the empirical distribution of neural LM predictions across contexts: out-of-distribution prediction in both RNNs and transformers involves a mix of global and local information, with the precise weighting of these two sources of information dependent on structural properties of the language being modeled. A natural next question is whether this weighting can be *controlled*: that is, whether modifications can be made to models or training procedures that affect the relative importance of the local and global hypotheses.

In this section, we explore *noise* as a possible source of this control. Models of both perceptual (local) noise and retrieval (global / contextual) noise play a key role in computational models of human sentence processing (Levy, 2008). In machine learning, various kinds of noise injected at training time—most prominently dropout (Srivastava et al., 2014), but also label noise and random word substitution and masking—are widely used as tools to regularize model training and limit overfitting. Here, we investigate whether these noising procedures qualitatively affect the *kind* of generalization behavior that neural LMs exhibit in the out-of-distribution contexts explored in Section 4.2.

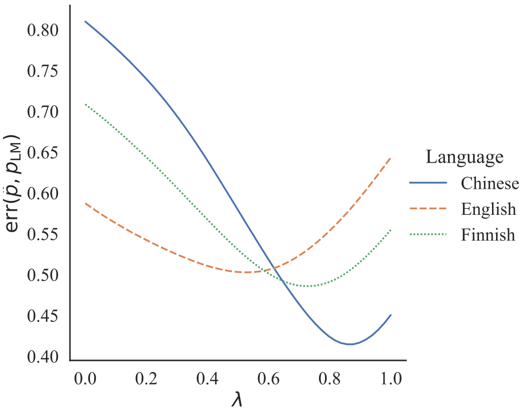

Figure 3: Effect of the log-linear interpolation parameter λ_1 (fixing $\lambda_2 = 1-\lambda_1$; Eq. (8)) when predicting out-of-distribution behavior in language models. As shown in Fig. 2, in English, an all-local hypothesis ($\lambda = 0$) is better than an all-global hypothesis ($\lambda = 1$), but true model behavior is best approximated by a log-linear combination of the two ($\lambda \approx 0.5$). Finnish and Chinese are also best approximated by an interpolation, but are closer to the global than the local hypothesis.

We investigate two kinds of noise: random word substitution and hidden state dropout. In all experiments, this noise is applied at training time only; model inference is run noiselessly when evaluating fit in Eq. (11). When computing $\tilde{p}_{\mathrm{G}}$ with Eq. (10) in these experiments, $p_{\mathrm{LM}'}$ is also trained without noise to approximate $\tilde{p}_{\mathrm{G}}$.

Random token substitution With probability p, input tokens are randomly replaced with samples from the unigram distribution. Random word substitution plays an important role in masking-based pretraining schemes (Devlin et al., 2018).

Hidden state dropout With probability p, features of context representations (RNN hidden states and transformer self-attention outputs) are randomly set to zero (Semeniuta et al., 2016).

Results are shown in Fig. 4. Across languages, word substitution modestly increases the predictive

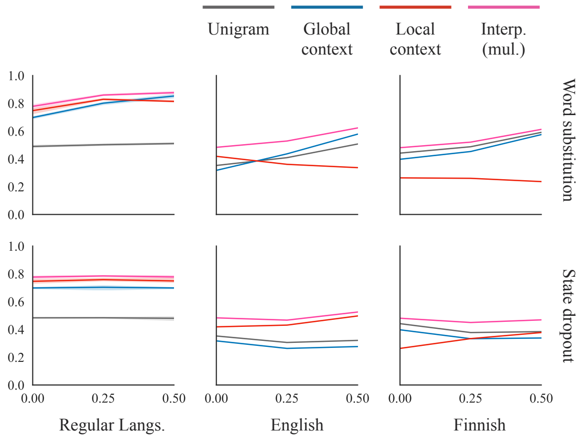

Figure 4: Accuracy $\mathrm{acc}(\tilde{p}, p_{\mathrm{LM}})$ of predicted generalization for English, Finnish, and regular-language GRUs when trained with token-swapping noise and state dropout noise. Token swapping sometimes improves the accuracy of the global context model, while state noising improves the accuracy of the local context model. Similar trends occur with Chinese; see Appendix A.

accuracy of the global context model, while sometimes decreasing the accuracy of the local context model. For natural languages, however, the improvement in the global context model is matched by improvements in the *unigram* baseline, indicating that these results may simply indicate decreased context-dependence. In regular languages, symbol-swapping noise increases the performance of the global hypothesis without improving the baseline, suggesting reliance on global information has actually increased. In all cases, state dropout improves the predictive accuracy of the local context model and often decreases the accuracy of both the global context model and the unigram baseline, suggesting indicating that state dropout encourages local generalization.

5 Explaining the experiments

With empirical evidence that interpolation between local and global models is a good approximation to out-of-distribution language model behavior, we next investigate whether this behavior can be explained theoretically. While we leave for future work a complete answer to this question, we conclude with the following proposition, which describes a set of conditions under which LM generalization will be well approximated by log-linear interpolation between $\tilde{p}_{\mathrm{L}}$ and $\tilde{p}_{\mathrm{G}}$.

Proposition 1. *Let θ be the parameters of a log-linear model optimizing:*

$$\arg\min_{\theta} \; - \sum_{X_{1:n}} \log p(X_n \mid X_{1:n-1}; \theta) + \lambda \|\theta\|^2 \qquad (13)$$

where $p(X_n \mid X_{1:n-1}) \propto \exp\{\theta_{X_n}^\top \phi(X_{1:n-1})\}$ and $\phi(\cdot)$ has an indicator feature for each value of X_{G}, X_{L}, and the conjunction $(X_{\mathrm{G}}, X_{\mathrm{L}})$. Suppose further that models with only *local or global features are boundedly worse than this model: specifically, that:*

$$\mathbb{E}|p(X_n \mid X_{\mathrm{G}}, X_{\mathrm{L}}) - p(X_n \mid \tilde{X})| < \epsilon \qquad (14)$$

uniformly for training *$\tilde{X}$ equal to either X_{G} or X_{L}. Then, in surprising contexts, $p(x_n \mid x_{1:n-1}; \theta)$ can be approximated by $\tilde{p}_\times$:*

$$\Big| p(X_n \mid X_{\mathrm{G}}, X_{\mathrm{L}}; \theta) - \tilde{p}_\times(X_n \mid X_{\mathrm{G}}, X_{\mathrm{L}}) \Big| < e^{4\epsilon/\lambda} - 1 \qquad (15)$$

where $\tilde{p}_\times(X_n \mid X_{\mathrm{G}}, X_{\mathrm{L}}) \propto \tilde{p}(X_n \mid X_{\mathrm{G}})\, \tilde{p}(X_n \mid X_{\mathrm{L}})$ and each $\tilde{p}(X_n \mid X_{\mathrm{L}} \text{ or } X_{\mathrm{G}})$ is an ℓ_2-regularized estimate of the corresponding distribution.

In other words, for log-linear language models with informative local and global features, the observed effectiveness of multiplicative interpolation is expected. Proof is given in Appendix C.

It is important to qualify this result in several ways: it relies on a feature function ϕ that may not be a realistic representation of the context feature produced by deep network models, involves strong assumptions about the independent predictive power of local and global features at training time, and becomes vacuous for large values of ϵ or small values of λ. It is weak in absolute sense ($e^{4\epsilon/\lambda}$ grows considerably faster than ϵ except for very large values of λ); its function is simply to relate predictions in surprising contexts to measurable properties of the training distribution. Nevertheless, the result shows that some aspects of interpolation behavior can be predicted from the parametric form of predictors alone; future work might strengthen this claim to more directly characterize the neural network predictors studied in this paper and explain the observed differences across languages.

6 Conclusion

When neural sequence models are exposed to out-of-distribution contexts with conflicting local and global information, their behavior can be predicted. Across natural and synthetic data distribution, sequence model generalization appears to be well approximated by either a local (n-gram-like) or global (skip-gram-like) predictor, and best approximated by a log-linear interpolation of the two whose weight can sometimes be controlled by noise-based regularization. This work suggests several avenues for future exploration: first, explaining *data-dependent* aspects of the local–global tradeoff (especially cross-linguistic differences that are not clearly explained by typological differences between languages); second, determining whether *architectural* improvements to standard sequence models can even more effectively target specific kinds of structured generalization.

References

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 conference on machine translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Christos Baziotis, Barry Haddow, and Alexandra Birch. 2020. Language model prior for low-resource neural machine translation.

Raki Bench. 2013. How to discover authentic pizza the Italian way. *Digital Nomad Travel Magazine*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The journal of machine learning research*, 3:1137–1155.

Ciprian Chelba and Frederick Jelinek. 1998. Exploiting syntactic structure for language modeling. In *ACL*.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jeffrey L Elman. 1990. Finding structure in time. *Cognitive science*, 14(2):179–211.

Dayne Freitag and Andrew McCallum. 1999. Information extraction with HMMs and shrinkage. In *Proceedings of the AAAI-99 workshop on machine learning for information extraction*, pages 31–36. Orlando, Florida.

Richard Futrell, Edward Gibson, and Roger P Levy. 2020. Lossy-context surprisal: An information-theoretic model of memory effects in sentence processing. *Cognitive science*, 44(3):e12814.

Richard Futrell and Roger Levy. 2017. Noisy-context surprisal as a human sentence processing cost model. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 688–698, Valencia, Spain. Association for Computational Linguistics.

Jon Gauthier, Jennifer Hu, Ethan Wilcox, Peng Qian, and Roger Levy. 2020. SyntaxGym: An online platform for targeted evaluation of language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 70–76.

Irving J Good. 1953. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(3-4):237–264.

Joshua T Goodman. 2001. A bit of progress in language modeling. *Computer Speech & Language*, 15(4):403–434.

David Guthrie, Ben Allison, Wei Liu, Louise Guthrie, and Yorick Wilks. 2006. A closer look at skip-gram modelling. In *LREC*, volume 6, pages 1222–1225. Citeseer.

Katri Haverinen, Jenna Nyblom, Timo Viljanen, Veronika Laippala, Samuel Kohonen, Anna Missilä, Stina Ojala, Tapio Salakoski, and Filip Ginter. 2014. Building the essential resources for Finnish: the turku dependency treebank. *Language Resources and Evaluation*, 48(3):493–531.

Gerard Manley Hopkins. 1918. The Wreck of the Deutschland. In Robert Bridges, editor, *Poems of Gerard Manley Hopkins*.

Frederick Jelinek and Robert Mercer. 1980. Interpolated estimation of Markov source parameters from sparse data. In *Proc. Workshop on Pattern Recognition in Practice, 1980*.

Slava Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE transactions on acoustics, speech, and signal processing*, 35(3):400–401.

Eugene Kharitonov and Rahma Chaabouni. 2020. What they do when in doubt: a study of inductive biases in seq2seq learners. *arXiv preprint arXiv:2006.14953*.

Dietrich Klakow. 1998. Log-linear interpolation of language models. In *Fifth International Conference on Spoken Language Processing*.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *1995 international conference on acoustics, speech, and signal processing*, volume 1, pages 181–184. IEEE.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *CoRR*, abs/1808.06226.

Thomas Kuhn, Heinrich Niemann, and Ernst Günter Schukat-Talamazzini. 1994. Ergodic hidden Markov models and polygrams for language modeling. In *Proceedings of ICASSP'94. IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages I–357. IEEE.

Brenden Lake and Marco Baroni. 2018. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning*, pages 2873–2882. PMLR.

Nur Lan, Emmanuel Chemla, and Roni Katzir. 2021. Minimum description length recurrent neural networks.

Roger Levy. 2008. A noisy-channel model of human sentence comprehension under uncertain input. In *Proceedings of the 2008 conference on empirical methods in natural language processing*, pages 234–243.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Adam Liška, Germán Kruszewski, and Marco Baroni. 2018. Memorize or generalize? searching for a compositional rnn in a haystack. *arXiv preprint arXiv:1802.06467*.

R Thomas McCoy, Robert Frank, and Tal Linzen. 2020. Does syntax need to grow on trees? sources of hierarchical inductive bias in sequence-to-sequence networks. *Transactions of the Association for Computational Linguistics*, 8:125–140.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*.

George A Miller and Jennifer A Selfridge. 1950. Verbal context and the recall of meaningful material. *The American journal of psychology*, 63(2):176–185.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajivc, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis M. Tyers, and Daniel Zeman. 2020. Universal dependencies v2: An evergrowing multilingual treebank collection. *ArXiv*, abs/2004.10643.

Adam Pauls and Dan Klein. 2012. Large-scale syntactic language modeling with treelets. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 959–968.

@pennstate452. 2021. https://twitter.com/pennstate452/status/1393635095202893826. Accessed: 2021-05-16.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Roni Rosenfeld. 1996. A maximum entropy approach to adaptive statistical language modeling.

Stanislau Semeniuta, Aliaksei Severyn, and Erhardt Barth. 2016. Recurrent dropout without memory loss. *arXiv preprint arXiv:1603.05118*.

Claude E Shannon. 1951. Prediction and entropy of printed english. *Bell system technical journal*, 30(1):50–64.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Amos Storkey. 2009. When training and test sets are different: characterizing learning transfer. *Dataset shift in machine learning*, 30:3–28.

Yee Whye Teh. 2006. A Bayesian interpretation of interpolated Kneser-Ney.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Noah Weber, Leena Shekhar, and Niranjan Balasubramanian. 2018. The fine line between linguistic generalization and failure in seq2seq-attention models. *arXiv preprint arXiv:1805.01445*.

Hui Zou and Trevor Hastie. 2005. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320.

A Noise Experiments in Other Settings

We include larger versions of the graphs from Figure 4 with a few other hypotheses added: the additive interpolation and random restart along with an *ignore* hypothesis that predicts LM generalization having consistent with having never seen the surprising token (i.e. using the predictive distribution from *before* the surprising token was observed).

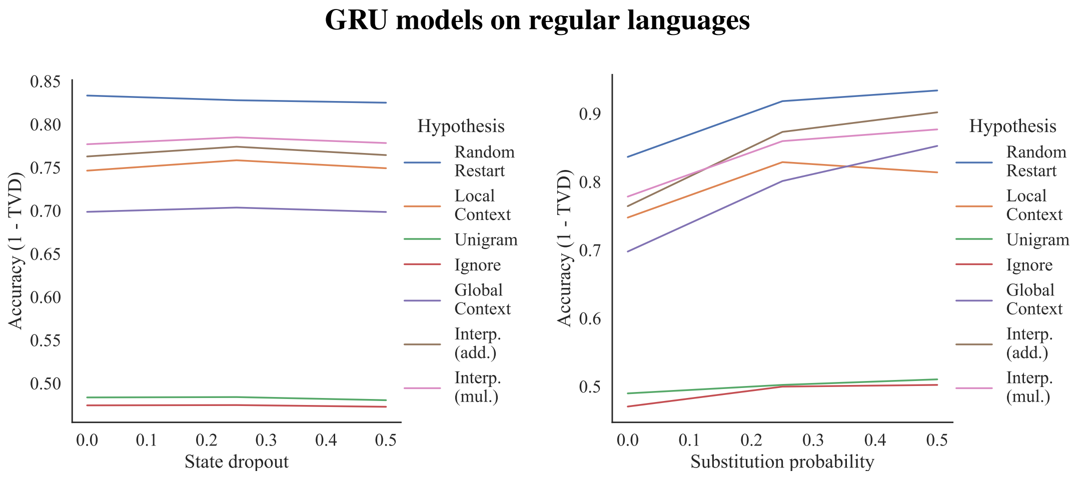

Figure 5: Detail version of graphs from Figure 4.

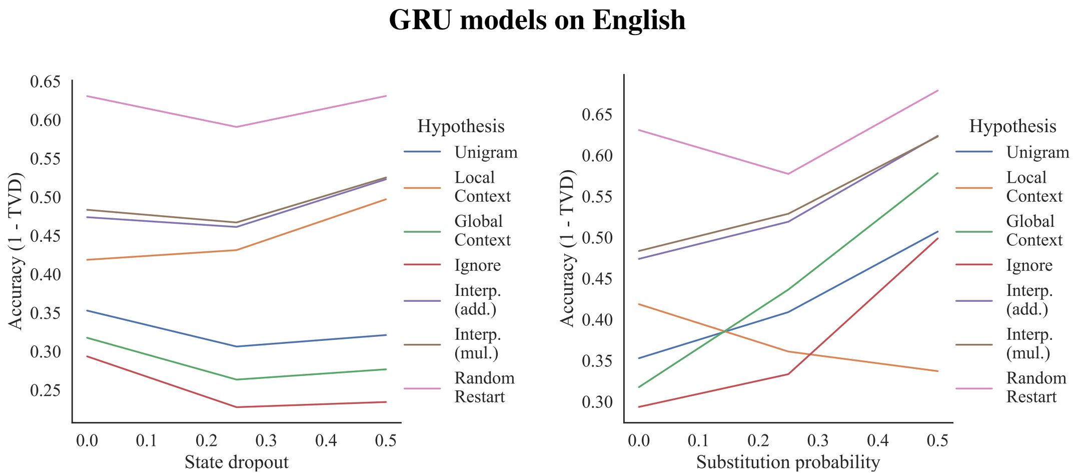

Figure 6: Detail version of graphs from Figure 4.

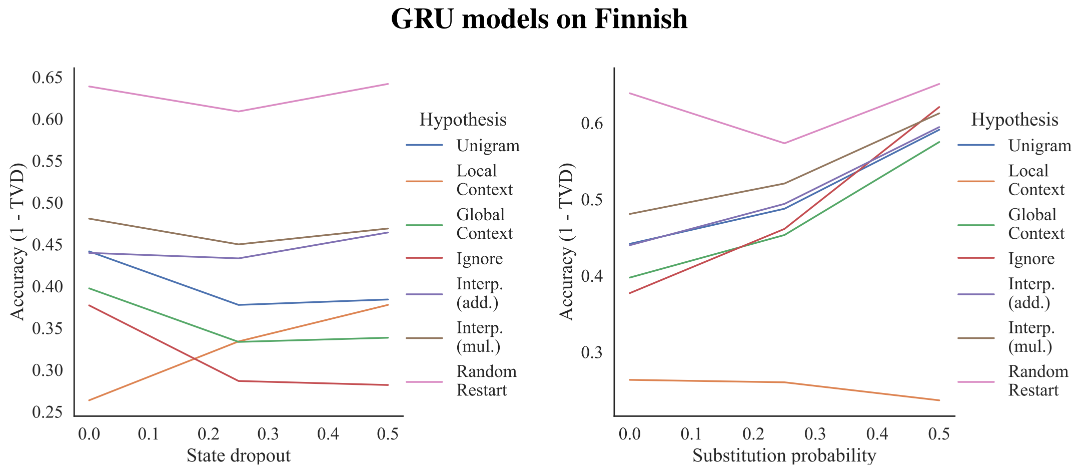

Figure 7: Detail version of graphs from Figure 4.

We include graphs of some settings for the noise experiments of Figure 4 that were omitted from the main paper for space reasons: Chinese GRU models and transformer models on regular languages. The trends are mostly similar.

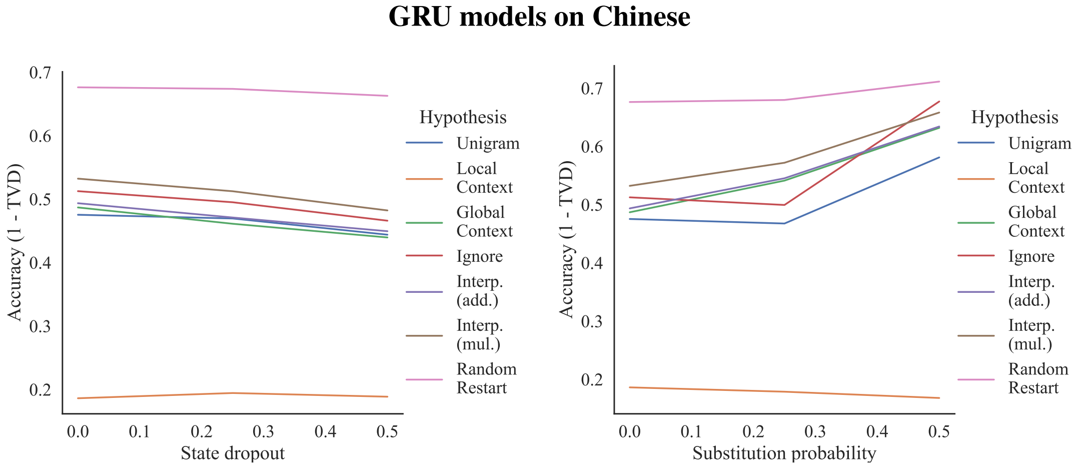

Figure 8: Graphs showing hypothesis performance for Chinese GRU language models trained under the same noise conditions as in Figure 4 and similar trends.

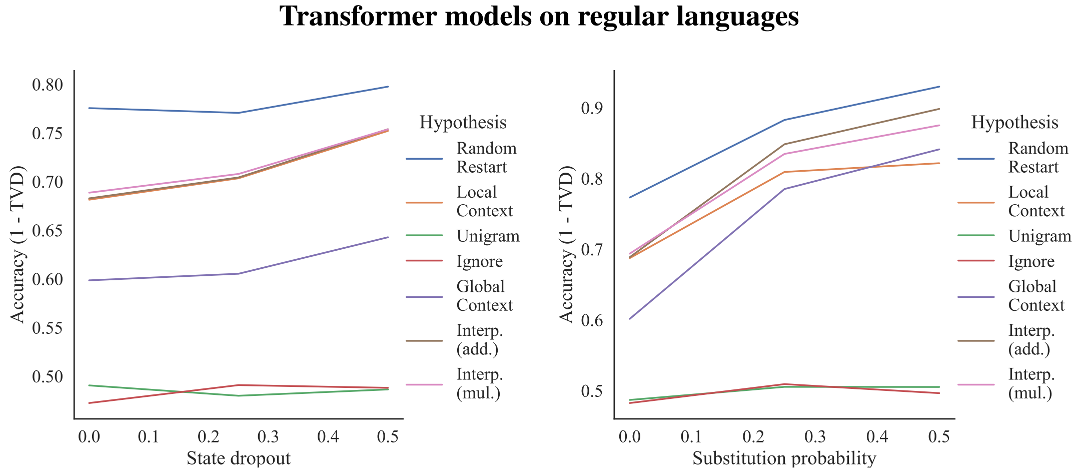

Figure 9: Graphs showing hypothesis performance for transformer language models for regular languages, trained under the same noise conditions as in Figure 4 and similar trends.

B Constructing Finite Automata

```
input:
    symbols: Set[char]
    states: Set[int]
    num_symbol_uses: int
    num_neighbors: int
    start_state: int

while true:
    symbol_counts = dict(s -> num_symbol_uses for s in symbols)
    dfa_edges = set()
    dfa_accepting_states = set()
    for state in states:
        neighbors = uniform(states).sample_without_replacement(n=num_neighbors)
        used_symbols = set()
        for neighbor in neighbors:
            while True:
                symbols_available = set(
                    s for s in symbols
                    if symbol_counts[s] > 0 and s not in used_symbols
                )
                if symbols_available.is_empty():
                    break
                symbol = uniform(available_symbols).sample()
                dfa_edges.add(edge(state, symbol, neighbor))
                used_symbols.add(symbol)
                symbol_counts[symbol] -= 1
                if used_symbols.size == symbols.size / states.size:
                    break
        if bernoulli(0.5).sample():
            dfa_accepting_states.add(state)

    dfa_edges = prune_unreachable_edges(dfa_edges, start_state)
    if dfa_edges.is_empty():
        continue

    return dfa
```

C Proof of Proposition 1

We begin with a simple lemma relating parameter weights in regularized log-linear models with mismatched feature sets.

Lemma 1. *Let ϕ^1 and ϕ^2 be feature distinct functions producing binary feature vectors, with $\phi^1_i(x) = \phi^2_i(x)$ for some i. Let $y_1, y_2, \ldots$ be output classes, and $\theta^1 = [\theta^1_{y_1}, \theta^1_{y_2}, \ldots]$ and θ^2 be the result of optimizing:*

$$\arg\min_\theta -\sum_{x,y} \log p(y \mid x; \theta) + \lambda \|\theta\|^2 \tag{16}$$

$$= \arg\min_\theta -\sum_{x,y} \theta_y^\top \phi(x) - \log \sum_{y'} \exp\{\theta_{y'}^\top \phi(x)\} + \lambda \|\theta\|^2 \tag{17}$$

for each $\phi = \phi^1, \phi^2$. As in Proposition 1, suppose the two feature functions produce similar predictors in the sense that:

$$\mathbb{E}|p(y \mid x, \theta^1) - p(y \mid x, \theta^2)| < \epsilon \tag{18}$$

for all training x. Then,

$$|\theta^1_{y,i} - \theta^2_{y,i}| \leq \frac{\epsilon}{\lambda} \tag{19}$$

Proof. Eq. (16) is convex, and at optimality its gradient with respect to each component of θ is 0. Then, for any $\theta_{v,i}$ we have:

$$0 = \sum_{x,y=v} \phi_i(x)(-1 + p(y \mid x; \theta)) + \sum_{x,y \neq v} \phi_i(x) p(y \mid x; \theta) + \lambda \theta_{v,i} \tag{20}$$

$$\theta_{y,i} = \frac{1}{\lambda}\Big(\sum_{x,y=v} \phi_i(x)(1 - p(y \mid x; \theta)) + \sum_{x,y \neq v} -\phi_i(x) p(y \mid x; \theta)\Big) \tag{21}$$

$$|\theta^1_{y,i} - \theta^2_{y,i}| = \frac{1}{\lambda} \sum_{x,y} \phi_i \big| p(y \mid x; \theta_1) - p(y \mid x; \theta_2) \big| \tag{22}$$

$$\leq \frac{\epsilon}{\lambda} \tag{23}$$

□

We can then obtain the main result:

Proof of Proposition 1. First note that we can estimate both local and global models using distributions of the form:

$$p(X_n \mid X_\mathrm{G}) \propto \exp\{\eta^\top \phi(X_\mathrm{G})\} \tag{24}$$

where $\phi(X_\mathrm{G})$ contains only global features. (and similarly for X_L). If these models are trained with the same regularization constant λ, the conditions of Lemma 1 will be satisfied with respect to each local or global feature. Then,

$$\Big| p(X_n \mid X_{1:n-1}; \theta) \;-\; (1/Z)\, p(X_n \mid X_\mathrm{G})\, p(X_n \mid X_\mathrm{L}) \Big| \tag{25}$$

$$= \left| \frac{\exp\{\theta_{X_n}^\top \phi(X_\mathrm{G}, X_\mathrm{L})\}}{\sum_v \{\exp \theta_v^\top \phi(X_\mathrm{G}, X_\mathrm{L})\}} - \frac{\exp\{\eta_{X_n}^\top \phi(X_\mathrm{G})\} \cdot \exp\{\mu_{X_n}^\top \phi(X_\mathrm{L})\}}{\sum_v \exp\{\eta_v^\top \phi(X_\mathrm{G})\} \cdot \exp\{\mu_{X_n}^\top \phi(X_\mathrm{L})\}} \right| \tag{26}$$

Here we have used θ to denote the parameters of the full model, η to denote the parameters of the global-only model, and μ to denote the parameters of the local-only model. Because each ϕ has an indicator for local, global, and (local, global) values, only two features are active in surprising contexts, corresponding to X_G and X_L. We will denote the indices of these weights i and j in each weight vector:

$$= \left| \frac{\exp\{\theta_{X_n,i} + \theta_{X_n,j}\}}{\sum_v \exp\{\theta_{v,i} + \theta_{v,j}\}} - \frac{\exp\{\eta_{X_n,i} + \mu_{X_n,j}\}}{\sum_v \exp\{\eta_{v,i} + \mu_{v,j}\}} \right| \tag{27}$$

Without loss of generality, assume the first term is larger than the second. By applying Lemma 1, we can rewrite θ in terms of η and μ. For shorthand, we will write $\Delta = \epsilon/\lambda$:

$$\leq \frac{\exp\{\eta_{X_n,i} + \Delta + \mu_{X_n,j} + \Delta\}}{\sum_v \exp\{\eta_{v,i} - \Delta + \mu_{v,j} - \Delta\}} - \frac{\exp\{\eta_{X_n,i} + \mu_{X_n,j}\}}{\sum_v \exp\{\eta_{v,i} + \mu_{v,j}\}} \tag{28}$$

$$= \Big(e^{4\Delta} - 1\Big) \frac{\exp\{\eta_{X_n,i} + \mu_{X_n,j}\}}{\sum_v \exp\{\eta_{v,i} + \mu_{v,j}\}} \tag{29}$$

$$\leq e^{4\epsilon/\lambda} - 1 \tag{30}$$

Note that this proof is not specific to local and global feature representations, but applies generically to *any* pair of regularized log-linear models with overlapping features and similar predictions. The proof is adapted from Theorem 1 of Zou and Hastie (2005), and we expect that it could be strengthened (as done there) to depend only on correlations between features of $\phi(X_\mathrm{G}, X_\mathrm{L})$ and each of $\{\phi(X_\mathrm{G}), \phi(X_\mathrm{L})\}$. □