Quantity doesn't buy quality syntax with neural language models

Marten van Schijndel
Cornell University
mv443@cornell.edu

Aaron Mueller
Johns Hopkins University
amueller@jhu.edu

Tal Linzen
Johns Hopkins University
tal.linzen@jhu.edu

Abstract

Recurrent neural networks can learn to predict upcoming words remarkably well on average; in syntactically complex contexts, however, they often assign unexpectedly high probabilities to ungrammatical words. We investigate to what extent these shortcomings can be mitigated by increasing the size of the network and the corpus on which it is trained. We find that gains from increasing network size are minimal beyond a certain point. Likewise, expanding the training corpus yields diminishing returns; we estimate that the training corpus would need to be unrealistically large for the models to match human performance. A comparison to GPT and BERT, Transformer-based models trained on billions of words, reveals that these models perform even more poorly than our LSTMs in some constructions. Our results make the case for more data efficient architectures.

1 Introduction

Recurrent neural network language models (LMs) can learn to predict upcoming words with remarkably low perplexity (Mikolov et al., 2010; Jozefowicz et al., 2016; Radford et al., 2019). This overall success has motivated targeted paradigms that measure whether the LM's predictions reflect a correct analysis of sentence structure. One such evaluation strategy compares the probability assigned by the LM to a minimal pair of sentences differing only in grammaticality (Linzen et al., 2016). In the following example, the LM is expected to assign a higher probability to the sentence when the verb agrees in number with the subject (1a) than when it does not (1b):

(1) a. The author laughs.
b. *The author laugh.

RNN LMs have been shown to favor the grammatical variant in the vast majority of cases sampled at random from a corpus (Linzen et al., 2016), but their accuracy decreases in the presence of distracting nouns intervening between the head of the subject and the verb, especially when those nouns are in relative clauses (Marvin and Linzen, 2018). Can we hope to address these deficits by training larger and larger networks on larger and larger corpora, relying on the "unreasonable effectiveness" of massive datasets (Halevy et al., 2009) and computational power?[1] Or would architectural advances be necessary to improve our LMs' syntactic representations (Kuncoro et al., 2018)?

This paper takes a first step towards addressing this question. We train 125 RNN LMs with long short-term memory (LSTM, Hochreiter and Schmidhuber, 1997) units, systematically varying the size of the training corpus and the dimensionality of the models' hidden layer, and track the relationship between these parameters and the performance of the models on agreement dependencies in a range of syntactic constructions (Marvin and Linzen, 2018). We also compare our RNNs' accuracy to that of GPT (Radford et al., 2018) and BERT (Devlin et al., 2019), Transformer-based LMs trained on very large corpora.

We find that model capacity does not consistently improve performance beyond a minimum threshold. Increased corpus size likewise has a moderate and inconsistent effect on accuracy. We estimate that even if training data yielded consistent improvements, an unreasonable amount of data would be required to match human accuracy. We conclude that reliable and data-efficient learning of syntax is likely to require external supervision signals or a stronger inductive bias than that provided by RNNs and Transformers.

[1] http://www.incompleteideas.net/IncIdeas/BitterLesson.html

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 5835–5841,
Hong Kong, China, November 3–7, 2019.

2 Language models

Architecture: All of the models we trained consisted of two LSTM layers. We trained models with 100, 200, 400, 800 or 1600 units in each hidden layer. Input and output weights were tied during training (Press and Wolf, 2017; Inan et al., 2017); consequently, the input embedding had the same dimensionality as the hidden layers.

Training data: We trained networks of each size on 2M, 10M, 20M, 40M and 80M words (2M = 2 million). We extracted five disjoint sections of the WikiText-103 corpus (Merity et al., 2016) for each corpus size;[2] in total, we trained 125 models (5 layer sizes $\times$ 5 corpus sizes $\times$ 5 corpus subsets).[3] We used the WikiText-103 validation set for validation.

Vocabulary: To ensure comparability across different models trained on different data, we used the same vocabulary for all the models we trained. The vocabulary consisted of an intersection of the 400k word GloVe vocabulary (Pennington et al., 2014) with the 50k words used by GRNN (see below); the resulting vocabulary had 28,438 words.

GRNN: We also report the syntactic performance of a publicly available LSTM LM (Gulordava et al., 2018, henceforth GRNN). This trained model has been the focus of a considerable amount of analysis work in the past year. The model has two layers of 650 units each, and was trained on 80M words.

Comparison with Transformers: Finally, we report results from two publicly available LMs based on non-recurrent self-attention (Transformers; Vaswani et al., 2017): GPT (Radford et al., 2018) and BERT (Devlin et al., 2019). Both of these models have been argued to learn powerful syntactic representations (Goldberg, 2019; Wolf, 2019). We compare our results to those reported by Wolf (2019) on a similar challenge set for these two Transformer models.[4]

GPT is a 12-layer Transformer with 110 million parameters (compared to GRNN's 39 million parameters); it was trained on 1 billion words. BERT has a similar architecture to GPT,[5] with three differences: it is bidirectional, it was trained on 3.3 billion words, and it has a different training objective than the typical LM: it attempts to predict a single masked word in a sentence given the words both before and after the target word. For comparability to the LSTMs and GPT, we examine the agreement performance of BERT when only the words before the target are given (in contrast to the bidirectional tests reported by Goldberg 2019).

3 Evaluation

We tested each trained model on the constructions from the Marvin and Linzen (2018) challenge set, which is based on the agreement paradigm described in the introduction.[6] We replaced the verbs used by Marvin and Linzen with the high-frequency verbs *is/are*, *was/were* and *has/have*. This was done to ensure that even the models trained on smaller corpora will have had exposure to both forms of the verb in question.

We performed statistical tests of our hypotheses using Bayes factors, which quantify the support the data provide for a more complex model compared to a simpler one (Rouder et al., 2009). We computed two-sample Bayes factors using `ttestBF` from the `BayesFactor` R package (Morey and Rouder, 2018) using default settings. Our null hypothesis was that there is no difference in accuracy between the two sets of models in question (e.g., all models with 400 units per layer compared to all models with 800 units per layer). The magnitude of the resulting Bayes factor K can be interpreted as follows (Jeffreys, 1961): $K < 1$ indicates that there is no difference in accuracy between the two model groups, and $K > 10$ provides strong evidence that the model groups obtain different accuracies.

4 Results

Increasing model size improved syntactic prediction accuracy up to 400 units per layer; further in-

[2] We made each of the 40M- and 80M-token training sets as disjoint as possible, but since Wikitext-103 only contains 103M tokens, it was not possible to make them wholly disjoint using Wikitext-103 as the mother corpus.

[3] Each model was initialized randomly and was trained to convergence with a dropout of 0.2 using a batch size of 20, backpropagating error for 35 observations. An initial learning rate of 20 was gradually annealed.

[4] The comparison is not exact because the Transformers were evaluated based on the rank of the two target verbs given the prefix, and the LSTMs based on the total log-probability of the sentence (including the final period); in addition, Wolf (2019) did not modify the dataset to use only high frequency verbs, as we describe in Section 3.

[5] BERT Base showed more accurate syntactic predictions than BERT Large (Goldberg, 2019), which has more parameters, so we only consider BERT Base.

[6] https://github.com/BeckyMarvin/LM_syneval

Corpus size		Layer size	
2M → 10M	**5508.8**	100 → 200	**768.5**
10M → 20M	0.1	200 → 400	**63.5**
20M → 40M	**12.9**	400 → 800	0.2
40M → 80M	0.2	800 → 1600	0.1

Table 1: Strength of evidence for improvements in agreement prediction accuracy as a result of increasing corpus size averaging across layer size (left) or layer size averaging across corpus size (right), as quantified by Bayes factors. Boldfaced Bayes factors indicate strong evidence of improvement.

creases in model size had no effect (see Table 1 for the statistical tests). Increasing the amount of training data impacted accuracy in an inconsistent way. Training on 10M tokens produced general improvements across all constructions compared to 2M tokens. Doubling the corpus to 20M did not affect accuracy, but doubling it again to 40M did. There was no evidence of further improvement between 40M and 80M words.

In the remainder of this section we analyze the effect of increasing model size and training corpus size on the models' predictions for each construction in the data set.[7] A subset of the results is shown in Fig. 1; for the full results, see Figs. 3, 4 and 5 in the Appendix.

Local number agreement: All models trained on 10M words or more obtained perfect or near-perfect accuracy (mean $> 99\%$) in the cases where the verb was adjacent to its subject: simple agreement (*the author has/*have books*) and agreement within a sentential complement (*the mechanics said the author has/*have books*). When trained on 2M words, the models performed slightly worse, but their accuracy was still very high (mean 95.6%), regardless of model size. Overall, we conclude that the plurality of specific nouns and the generalization that a verb has to agree with a noun can be learned very quickly.

Attractors: Agreement across subject relative clauses (*the author that likes the guards has/*have books*) and across prepositional phrases (*the author next to the guards has/*have books*; Fig. 1a) benefited from increasing the hidden layer size to 400, but showed little improvement when hidden layer size was increased further. Accuracy in these constructions consistently improved as the amount of training data increased.

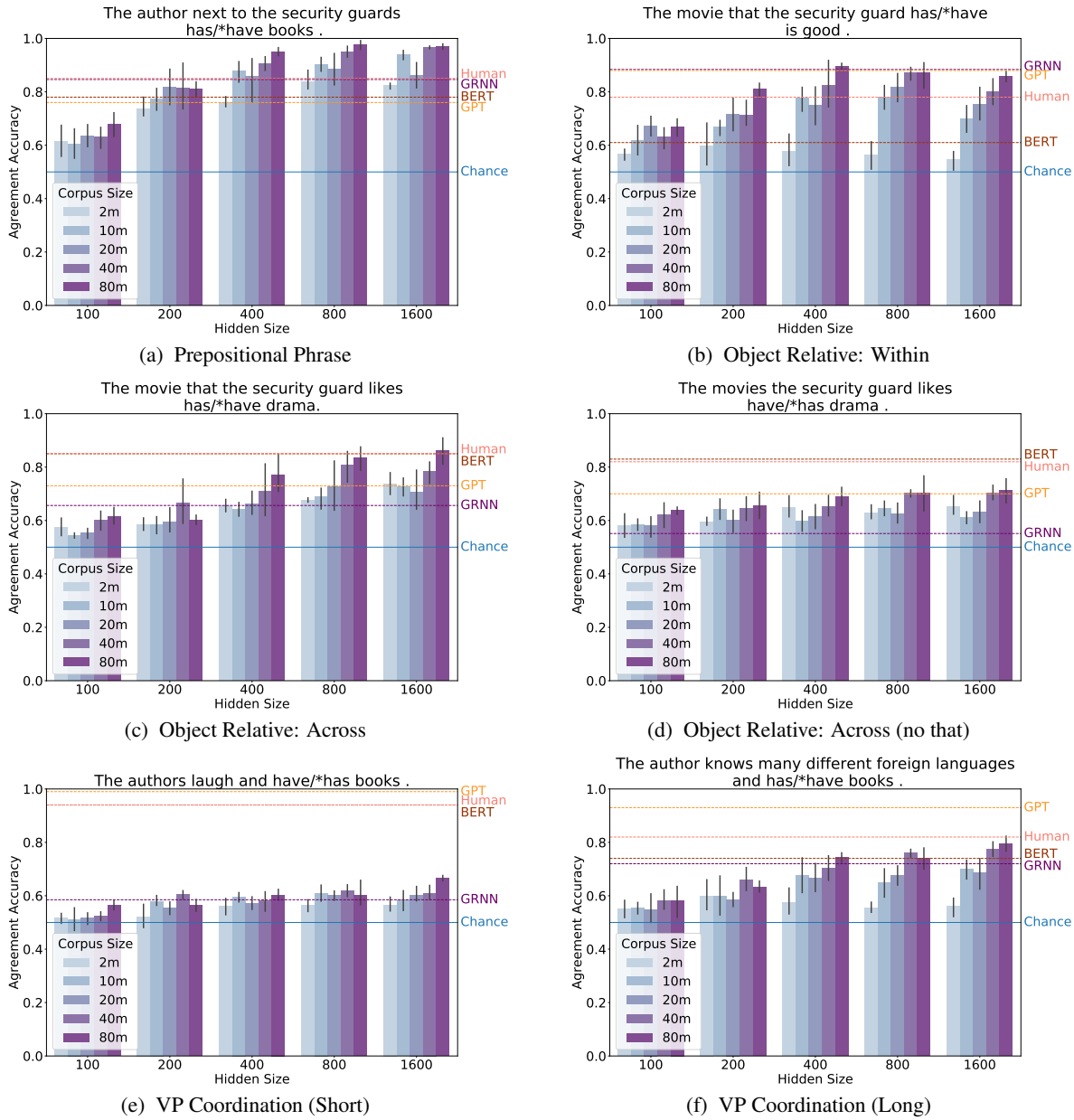
Object relative clauses: Expanding the training corpus improved local agreement **within** object relative clauses (*the movies that the guard has/*have are good*; Fig. 1b) for all model sizes, but only improved agreement **across** those clauses (*the movie that the guards like has/*have drama*; Fig. 1c) in models with larger hidden layers. Larger hidden layers improved accuracy in object relatives only when a relativizer was present, and only up to about 80% accuracy. When a sentence lacked an overt relativizer (*the movies the security guard has/*have are good*; Fig. 1d), all models performed poorly, with accuracy levelling off around 70%.

Coordination: Perhaps surprisingly, all LSTM LMs struggled with agreement in a coordinated verb phrase (*the authors laugh and have/*has books*; Fig. 1e), even though this construction does not include distracting nouns between the subject and the second verb. In larger models trained on more data, accuracy was higher when the second verb was **further** from the subject (*the authors know many different languages and have/*has books*; Fig. 1f).

Training on more than 10M tokens did not improve accuracy in short VP coordination, even when the amount of data was multiplied by eight (10M → 80M: $K < 1$), unlike coordination across long VPs, which benefited from additional data (10M → 80M: $K > 90$). These results further challenge the assumption that increased amounts of training data will lead to adequately abstract syntactic representations: RNNs show a limited ability to generalize from instances of a construction that have longer constituents to instances with shorter constituents.

Reflexive anaphora: A reflexive pronoun such as *themselves* must have an antecedent with the appropriate plurality in the same clause as the pronoun (*The manager that the architects like doubted himself/*themselves*). Accuracy was not strongly affected by the parameters we varied: reflexive agreement accuracy across a relative clause was consistently mediocre (61%–76%) regardless of model size or the amount of training data.

[7] See Table 2 in the Appendix for a Bayes factor analysis of the improvement in each construction for each amount of training data.



(a) Prepositional Phrase

(b) Object Relative: Within

(c) Object Relative: Across

(d) Object Relative: Across (no that)

(e) VP Coordination (Short)

(f) VP Coordination (Long)

Figure 1: LSTM agreement performance in several syntactic constructions. The solid horizontal line indicates chance performance. The dashed lines show the performance of GPT and BERT as reported by Wolf (2019), the performance of humans as reported by Marvin and Linzen (2018), and the performance of GRNN. Error bars reflect standard deviation across the five models in each category.

Transformers: Despite having more parameters and having been trained on significantly larger corpora, the two Transformer models performed either as well as or more poorly than our LSTMs in seven of the ten subject-verb agreement conditions. BERT underperformed GPT in several conditions despite having been trained on three times as many tokens as GPT.[8]

5 How much data would be enough?

How much training data would be required for an LSTM LM to perform at a human level (as reported by Marvin and Linzen 2018) in the condi-

[8] Goldberg (2019) reports much better results using a setup in which BERT has access to both left and right context. We hypothesize that the task is made significantly simpler when the model knows where the target word is relative to the end of the sentence. For example, if the point of prediction is at the last word of the sentence, it is also the last point at which the verb agreeing with the main clause subject could possibly occur; the model does not need to detect the end of the relative clause to perform the task in this case.

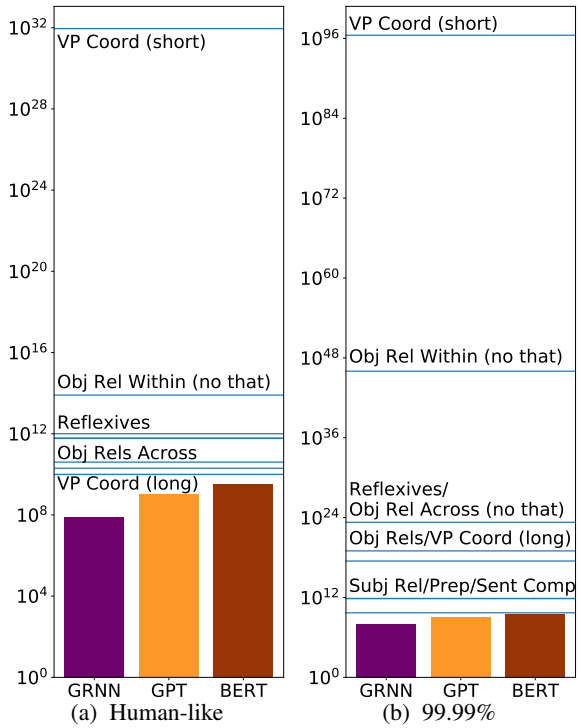


Figure 2: Lines depict number of training tokens needed for LSTMs to achieve human-like (left) or 99.99% accuracy (right) in each syntactic agreement condition, according to our estimates. Bars depict the amount of data on which each model was trained.

tions in which our models do not already perform at a human level? As a conservative estimate, we measured the error reduction achieved by doubling the data from 20M to 40M tokens (the largest error reduction we observed beyond 2M $\rightarrow$ 10M).[9] Under the assumption that each subsequent doubling of training data would produce the same percent error reduction, we predicted the amount of data required to obtain human-like and 99.99% accuracy (see Fig. 2).[10] We found that every remaining construction would require over 10 billion tokens to achieve human-like performance, and most would require trillions of tokens to achieve perfect accuracy – an impractically large amount of training data, especially for these relatively simple syntactic phenomena.

[9]See Tables 3 and 4 in the Appendix for data requirements estimated from other error reduction rates.

[10]Human performance on this task is well known to be far from perfect, with error rates approaching 25% in some contexts (Bock and Miller, 1991). While modeling human errors is of considerable interest to cognitive scientists (Linzen and Leonard, 2018), we believe that in most applied contexts it is desirable for the model to make no errors at all.

6 Discussion

We have investigated the effect of network size and training corpus size on the quality of the syntactic representations of LSTM LMs, as measured by agreement prediction accuracy. Increased model size had limited benefits; models with 400 hidden units performed significantly better than smaller models, but further increases in network size had no effect. The limited effect of network size is consistent with previous findings on sequence labeling tasks (Reimers and Gurevych, 2017; Greff et al., 2017). We have also shown that increasing the amount of training data is unlikely to result in human-like accuracy in all cases.

We found a striking difference in agreement accuracy between short and long coordinated verb phrases: performance on short phrases was poorer. While RNNs are known to struggle with generalizing short patterns to longer sequences, this pattern constitutes a failure to generalize to *shorter* sequences (cf. Trask et al., 2018); techniques for improving longer distance dependency learning in LMs (e.g., Trinh et al., 2018; Dai et al., 2019) are unlikely to mitigate this deficit. This suggests that challenge sets should include materials that can be used to ascertain whether the model's syntactic representations are robust to syntactically irrelevant factors such as constituent length.

GPT and BERT, Transformer models trained on very large corpora, did not consistently outperform the LSTMs trained on several orders of magnitude less data. Other studies suggest that Transformer models suffer from similar problems as the LSTMs we have analyzed. BERT's agreement accuracy decreases as the subject becomes more distant from its verb (Bacon and Regier, 2019). Dramatically increasing the pre-training corpus for a BERT-like model from 562M words to 18G words only leads to a modest improvement in its natural language inference accuracy, from 81.7% to 82.3% (Baevski et al., 2019). Overall, this body of results points to the limited data efficiency of standard RNNs and Transformers, and indicates that learning syntax from realistic amounts of data—in particular the amount of data available to humans when they learn language—may require syntactically structured architectures or explicit syntactic supervision (Enguehard et al., 2017; Kuncoro et al., 2018, 2019; Wilcox et al., 2019).

References

Geoff Bacon and Terry Regier. 2019. Does BERT agree? Evaluating knowledge of structure dependence through agreement relations. Technical report, UC Berkeley.

Alexei Baevski, Sergei Edunov, Yinhan Liu, Luke Zettlemoyer, and Michael Auli. 2019. Cloze-driven pretraining of self-attention networks. Technical report, Facebook AI Research.

Kathryn Bock and Carol A. Miller. 1991. Broken agreement. *Cognitive Psychology*, 23(1):45–93.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. Technical report, Carnegie Mellon University.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Émile Enguehard, Yoav Goldberg, and Tal Linzen. 2017. Exploring the syntactic abilities of RNNs with multi-task learning. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 3–14. Association for Computational Linguistics.

Yoav Goldberg. 2019. Assessing BERT's syntactic abilities. Technical report, Bar Ilan University.

Klaus Greff, Rupesh K. Srivastava, Jan Koutník, Bas R. Steunebrink, and Jürgen Schmidhuber. 2017. LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *Proceedings of the 2018 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Alon Halevy, Peter Norvig, and Fernando Pereira. 2009. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hakan Inan, Khashayar Khosravi, and Richard Socher. 2017. Tying word vectors and word classifiers: A loss framework for language modeling. In Yoshua Bengio and Yann LeCun, editors, *Proceedings of the Fifth International Conference on Learning Representations*. International Conference on Learning Representations.

Harold Jeffreys. 1961. *Theory of Probability*, 3rd edition. Oxford University Press, Oxford.

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*.

Adhiguna Kuncoro, Chris Dyer, John Hale, Dani Yogatama, Stephen Clark, and Phil Blunsom. 2018. LSTMs can learn syntax-sensitive dependencies well, but modeling structure makes them better. In *Proceedings of the 2018 Annual Meeting of the Association for Computational Linguistics*, pages 1426–1436. Association for Computational Linguistics.

Adhiguna Kuncoro, Chris Dyer, Laura Rimell, Stephen Clark, and Phil Blunsom. 2019. Scalable syntax-aware language models using knowledge distillation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3472–3484, Florence, Italy. Association for Computational Linguistics.

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.

Tal Linzen and Brian Leonard. 2018. Distinct patterns of syntactic agreement errors in recurrent networks and humans. In *Proceedings of the 2018 Annual Meeting of the Cognitive Science Society*, pages 690–695. Cognitive Science Society.

Rebecca Marvin and Tal Linzen. 2018. 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. Association for Computational Linguistics.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Wikitext-103. Technical report, Salesforce.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH 2010)*, pages 1045–1048, Makuhari, Chiba, Japan.

Richard D. Morey and Jeffrey N. Rouder. 2018. *BayesFactor: Computation of Bayes Factors for Common Designs*. R package version 0.9.12-4.2.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of EMNLP*.

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. In *Proceedings of the 2017 Annual Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. Technical report, OpenAI.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report, OpenAI.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348. Association for Computational Linguistics.

Jeffrey N. Rouder, Paul L. Speckman, Dongchu Sun, Richard D. Morey, and Geoffrey Iverson. 2009. Bayesian t-tests for accepting and rejecting the null hypothesis. *Psychonomic Bulletin & Review*, 16(2):225–237.

Andrew Trask, Felix Hill, Scott E Reed, Jack Rae, Chris Dyer, and Phil Blunsom. 2018. Neural arithmetic logic units. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 8035–8044. Curran Associates, Inc.

Trieu H. Trinh, Andrew M. Dai, Minh-Thang Luong, and Quoc V. Le. 2018. Learning longer-term dependencies in RNNs with auxiliary losses. In *Proceedings of the 35th International Conference on Machine Learning*, pages 4965–4974. PMLR 80.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Ethan Wilcox, Peng Qian, Richard Futrell, Miguel Ballesteros, and Roger Levy. 2019. Structural supervision improves learning of non-local grammatical dependencies. In *Proceedings of the 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics.

Thomas Wolf. 2019. Some additional experiments extending the tech report "assessing BERT's syntactic abilities" by Yoav Goldberg. Technical report, Huggingface Inc.