A Replication Study to Assess the Ability of LSTMs to Learn Syntax-Sensitive Dependencies

Suhas Dara
University of Texas at Austin
Department of Computer Science
suhasdara@utexas.edu

Mofei Zhang
University of Texas at Austin
Department of Computer Science
mofei.zhang@utexas.edu

Abstract

We investigate whether recurrent neural networks (RNNs, LSTMs) suffice to learn and represent non-local, context-dependent syntactic structures (paper reproduction of Linzen et al. (2016)). In addition, (1) we take inspiration from Gulordava et al. (2018) and evaluate whether LSTMs are able to provide accurate predictions in the absence of collocational information; (2) we probe models trained under direct supervision for other syntactic structures such as POS tags. We choose to reproduce tasks based on the same natural language structure, subject-verb agreement.

1 Introduction

Recurrent Neural Networks (RNNs) excel at learning statistical probabilities in sequences and relations between adjacent words. Even naive feed-forward networks trained on bag-of-word features can achieve significant accuracy rates on some problems. However, many natural language tasks require learning non-local and/or hierarchical structures such as dependency trees and part-of-speech tags. In this project, we investigate whether RNNs can successfully learn these tasks, even though they do not explicitly encode syntactical structures.

We pursue a paper replication of Linzen et al. (2016). We re-implement all model, training, and evaluation code in PyTorch[1], loosely based on a Tensorflow implementation provided by Linzen et al.[2] We show that we were able to reproduce most results achieved by Linzen et al. (due to limited compute resources, we did not reproduce the subsection that evaluates the Google LM (Józefowicz et al., 2016)).

[1]Our code is available here: https://github.com/mofeiZ/nlp_final_proj

[2]Linzen et al. code is available here: https://github.com/TalLinzen/rnn_agreement

Extensions to paper reproduction We probe trained models to determine whether they learned language structure or only relied on collocational information (e.g. frequently co-occurring nouns and noun phrases). We aim to understand if LSTMs can still learn sentence structure in the absence of co-occurring nouns.

In addition, we probe models trained under direct supervision (for subject-verb agreement) to determine whether they implicitly learned to represent other natural language structure.

2 Background

2.1 Subject verb agreement

A grammatically correct English sentence must have its subject(s) and verb(s) agreeing in number. That is, if a subject is singular or plural, then its verb must be singular or plural respectively. The model needs to understand the syntactic structure of the sentence to be able to predict the verb. This cannot be done utilizing models that only utilize a fixed-size or only the neighborhood of the verb because of several different reasons:

Varying distance: A fixed-size model such as an n-gram model cannot be used to capture the syntactic structure of a sentence. This is because the distance between the noun and the verb is not in a fixed window e.g.

> The **book** on the shelf **is** about dogs.
> The **book** on the shelf in the library **is** about dogs.

Simple RNNs can theoretically capture the syntactic structure over sentences of varying length, however, due to vanishing gradients (Hochreiter, 1998), they may not yield consistent results over longer sentences. LSTMs (with memory gates) should perform better for this task.

Intervening nouns: The noun and verb pair may not immediately follow each other. As the complexity of a sentence increases, there is a large probability of intervening nouns in between the noun and its corresponding verb. Intervening nouns are nouns that are not associated with the verb in the sentence. These intervening nouns may or may not have the same number as the subject noun. In the case that the nouns disagree with the subject noun, they are called agreement attractors, as they steal the noun-verb agreement from the original subject noun. Looking at the local context may lead to a wrong prediction. As an example:

> The **books** over there **are** about dogs.
> The **books** on the shelves **are** about dogs.
> The **books** on the shelf **are** about dogs.

In the first example, there is no intervening noun. In the second example, the intervening noun is of the same number as the subject noun, but in the third example, the intervening noun is an agreement attractor.

Relative clauses: The intervening nouns can be further complicated if they form a relative clause within the sentence. Relative clauses may seem like the correct attractor of the verb agreement but are not. As an example:

> The **books** *of* the student ...
> The books *that* the **student** ...

In the first example, there is a prepositional phrase (PP) that begins with the word "of". This does not change the active subject of the sentence. However, in the second example, there is a relative clause (RC) that begins with the word "that". This does change the active subject of the sentence within the relative clause context.

2.2 Related work

This work is a direct paper replication of Linzen et al. (2016), whose contents are explained throughout this paper. Linzel et al. were the first to analyze whether RNNs, specifically LSTMs, had the capacity to implicitly learn and represent natural language structure.

A followup work by Gulordava et al. (2018) showed that the supervised learning models by Linzen et al. memorized some word co-frequencies. Gulordava et al. generated sentences with random nouns in multiple languages for training and evaluation. Results of this work strongly support the results of Linzen et al.: LSTMs can learn syntactic structure with direct supervision.

Another followup work by Kuncoro et al. (2018) added explicit structure in the form of constituency parse trees to the input data of the language modeling task. This model gives much higher accuracy when probing for subject-verb agreement.

3 Methods

3.1 Data

We use the dataset created by Linzen et al. The corpus of example sentences and POS labels was created from Wikipedia articles. Each example contains sentence, target noun/verb, and additional information to enable a variety of experiments. In the data, words with low frequency were replaced with their corresponding part-of-speech (POS) tags[3] to prevent the models from learning outlier patterns.

In each task, we only use sentences as model inputs. Our dataset split ratios is adopted from Linzen et al. for consistency. The data consists of 1,577,211 examples of which 1,419,490 (90%) examples are reserved for extensive testing. The remaining data is split into 141,949 (9%) examples for training and 15,772 (1%) examples for validation.

3.2 Supervised learning

We reproduce the Number Prediction task. In this task, the model is trained to predict the number (singular or plural) of a present-tense verb based on the preceding words. The sentence below is an example of a training example with the label "singular".

> The **book** on the shelf _____

To understand whether LSTMs can capture the syntactic context of a sentence, we consider variations of the Number Prediction task and establish baselines similar to Linzen et al. (2016) We group our baselines into two categories. These aim to understand if the model is using syntactic context primarily or secondarily to the subject noun and the verb itself.

3.2.1 Noun-only baselines

All sentence context information is withheld except for the subject noun, the verb, and all the interven-

[3] Penn Treebank tags

ing nouns. As other parts of speech are unavailable in the context, the model will not learn any context. Another variation of this baseline is one where only the POS tags are available for the nouns instead of the actual nouns.

3.2.2 Grammaticality baselines

Sentence context is available, but the subject noun and the verb themselves may not agree grammatically. In one variation, the verb seen during training is always singular and as a result, may or may not agree with the subject noun. In another variation, the verb seen during training always has the opposite number to the subject noun and, as a result, always disagrees with the subject noun.

3.3 Self-supervised learning

We reproduce the Language Modeling (LM) task. This type of self-supervised learning is used to train many large-scale models today due to the sheer amount of training examples that could be generated (Vaswani et al. (2017), Devlin et al. (2018)). In this task, a model is trained to predict the next word given previous words, with no syntactic annotations. In the absence of structured data, LSTM models trained on the LM task should learn to represent meaningful structure in the latent space.

After training our model on the LM objective, we probe whether it has learned the necessary syntax for subject-verb agreement. This is done by evaluating the following condition on sentences in the test dataset.

$$\Pr[\text{verb that agrees with subject} \mid \text{prev words}] > \Pr[\text{verb that disagrees with subject} \mid \text{prev words}]$$

Due to limited compute resources, we were only able to train and evaluate the LM model on one-fourth of the provided dataset.

3.4 Extension: Number Prediction with randomized subjects

We expand on the paper reproduction by evaluating our models on sentences with randomly generated subjects. This is inspired by the work of Gulordava et al. (2018) which shows that LSTMs can still learn sentence structure in the absence of co-occurring nouns.

We make modifications to the dataset provided by Linzen et al. First, we discover the sets of all singular and plural nouns in the provided dataset, then we replace every subject with a randomly sampled noun. This is slightly different than the methodology used by Gulordava et al., which replaces all nouns in the sentence with randomly sampled ones. We believe that we can achieve similar results by replacing only the sentence subjects.

This modified dataset is used to probe a model trained with the original dataset on the Number Prediction task. In addition, we train and evaluate another model using the modified dataset with the goal of reproducing the results given by Gulordava et al.

3.5 Extension: Probing Verb Number Prediction for POS

We expand on the paper reproduction by probing models trained on the Verb Number Prediction task for POS information. Our goal is to show that models trained on subject-verb agreement with direct supervision can implicitly capture POS information.

Our probe model is a simple single layer feed-forward network that takes the output of the last hidden layer from our Number Prediction LSTM model and predicts POS tags (45 classes total from the Penn Treebank).

3.6 Model

The model architecture for the Number Prediction task was adapted from the model created by Linzen et al. The words are encoded as one-hot vectors and then embedded using 50-dimensional word embeddings, which are trained and not fixed. The word embeddings outputs are passed through an LSTM with 50 hidden units and the final state of the LSTM is passed through a linear layer. The model is trained using Binary Cross Entropy (BCE) loss and optimized using Adam. Additionally, an early stopping mechanism is set in place to stop training once validation loss starts increasing again.

All baselines in Section 3.2 and the Number Prediction task are trained with the aforementioned model architecture. The models were set to train for 10 epochs but usually early stopped before completing 10 epochs. When early stopping was removed, no significant improvement was observed.

Our LM model is almost identical to the aforementioned model architecture, only training with Cross Entropy Loss instead of BCE. The architecture of our probe model is discussed in Section 3.5.

We only evaluate models on tasks that they are trained for. In our discussion of results, we shorten

Model	Accuracy	F1-score
Number prediction	98.81	98.14
Only Nouns baseline	95.00	92.10
POS Nouns baseline	94.84	91.95
Always Singular Verb baseline	98.79	98.11
Reversed Grammaticality baseline	95.93	95.93

Table 1: Accuracy and F1-score results for all the baselines models and the number prediction model.

"LSTM model trained on the ___ task" to the "___ model".

4 Results and Analysis

Overall, the accuracy and F1-score of the Number Prediction task are promising, as seen in Table 1. The LSTM model demonstrates a 98.81% accuracy and a 98.14% F1-score. The model shows a 1.19% error rate on predictions.

The noun-only baselines have a significantly higher error rate at 5.00% for only nouns, and 5.16% for POS tags of the nouns. We will explore this further in Section **??**, but it suggests that syntactic information from the context of the sentence helps the LSTM make better predictions.

When comparing with the grammaticality baselines, it is not immediately clear whether the number prediction model is overly dependent on using the subject noun and the verb itself to make its predictions. For the Always Singular Verb model, there is no significantly higher error rate at only 1.21%. However, the Reversed Grammaticality model has a higher error rate at 4.07%. In section 4.3, we will investigate where the error in these baselines lies.

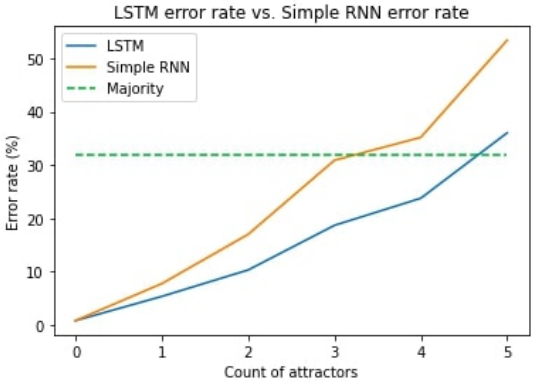


Figure 1: Verb number prediction error rate depending on the number of agreement attractors and the type of the model

4.1 Comparison to Simple Recurrent Neural Networks

To truly understand how much of the model's success should be attributed to the LSTM cell, we first repeat the Number Prediction task using a Simple Recurrent Neural Network (RNN), specifically an Elman network (Elman, 1990). The performance of the LSTM network was consistently better than the Elman network by a factor of about 1.5, as seen in Figure 1. However, this doesn't necessarily imply that the Elman network is unable to capture the same syntactic structures captured by the LSTM network. One possibility is the limitation of Elman networks to learn structures over long sentences due to vanishing gradients (Hochreiter, 1998). As the number of agreement attractors increase in a sentence, the length of the sentence is also likely to increase. This could explain the Elman network's worse performance on these examples. So, for the remainder of the results, we only consider the performance of LSTMs over Elman networks.

4.2 Distance

We first look at whether the model consistently makes correct predictions as the word distance between the subject noun and the verb increases. Similar to Linzen et al., we do not consider examples where there are intervening nouns (whether of the same type or not). This is to avoid introducing an extra variable. The model performs well even at larger distances. As seen in Figure 2, the model has error rates less than 1% when the subject noun and verb are adjacent and there is not a very large reduction in performance even for distances of 12-14. It is important to consider that most sentences in the English language have smaller distances, which is why the overall error rate is not very high.

4.3 Agreement attractors

We next assess whether intervening nouns between the subject noun and the verb played a role in the model's error rates. We check how the number

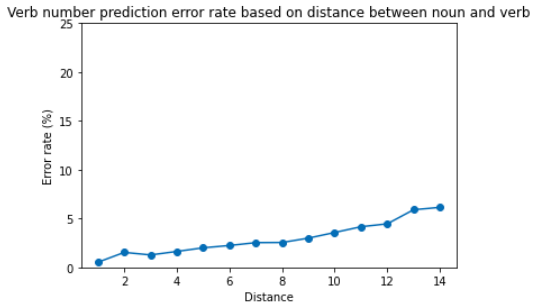


Figure 2: Verb number prediction error rate depending on the distance between the noun and the verb excluding examples containing intervening nouns

of intervening nouns affects the error rate in the number prediction model compared to the various baseline models, and also if there are any significant differences between singular and plural nouns.

To analyze whether the number of intervening nouns between the subject noun and the verb, we eliminate some variables similar to Linzen et al. We only use examples with all intervening nouns of the opposite number to the subject noun. These opposite number intervening nouns are labeled as agreement attractors as mentioned in Section 2.1. As all the intervening nouns are of the same number, this is a homogeneous intervention.

Figure 3a shows the comparison between the Number Prediction model and the noun-only baseline models as the number of agreement attractors increase between the subject noun and the verb. While the Number Prediction model's error rate stays below the majority label even with 4 homogeneous agreement attractors, the noun-only baseline models struggle even with a single agreement attractor. With 4 agreement attractors, the error rate for the Only Nouns model is 88.39% and for the POS Nouns model is 74.00%. The POS Nouns model performs slightly better, potentially due to seeing a lesser number of distinct tokens, but both baseline models perform much worse than the Number Prediction model (error rate of 23.74%). This demonstrates that syntactic information is necessary for the model to perform well, which is unavailable in the noun-only baselines.

Figure 3b shows the comparison between the Number Prediction model and the grammaticality baseline models. The error rate remains below majority up to 4 agreement attractors and quite close to each other for the Number Prediction model and both the grammaticality baseline models. This probably means that the model is using the syntactic context a lot more than the subject noun and the verb themselves, especially in examples where there is a lesser number of agreement attractors. However, looking at overall accuracy results, the Reversed Grammaticality model performed slightly worse than the Singular Verb Always model and the Number Prediction model, likely because of the slightly higher error rates overall, and potentially because it performed worse at larger number of agreement attractors.

Figure 4 shows that the error rate in predicting the verb number is not more prevalent in one number than the other. Both singular and plural nouns have similar error rates regardless of the number of the agreement attractors between the subject noun and the verb. This means that the Number Prediction model is not biased towards a single type of noun or verb. The errors from the model are split across both types of nouns and verbs.

4.4 Relative clauses

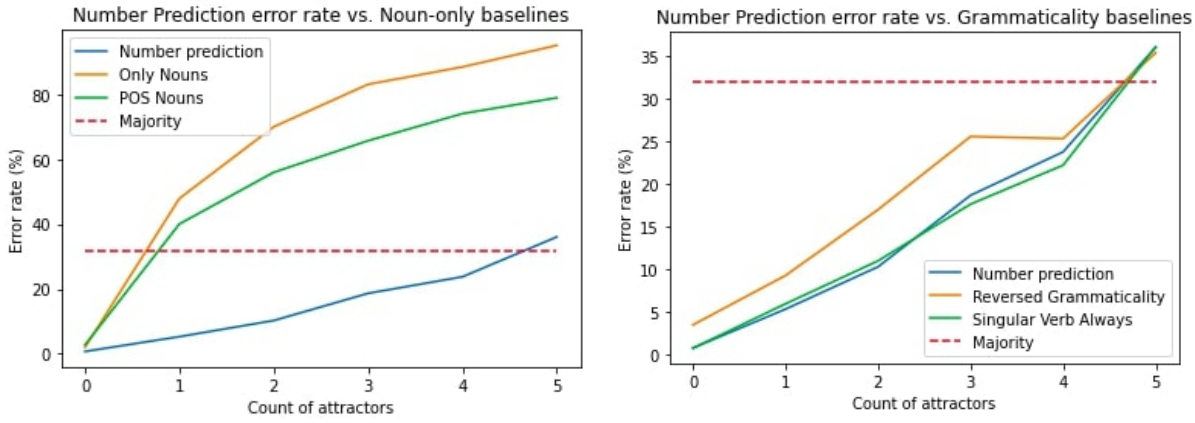
Relative clauses contain intervening nouns and can typically be challenging for a model because the intervening noun is also often accompanied by a verb that agrees with that intervening noun. If this intervening noun is also an agreement attractor, the model may be misguided by the intervening nouns and verbs. When the error rate of the model was considered on examples with a single agreement attractor where a relative clause was present or not, we observed that the error rate on examples without a relative clause was 4.40%, while the error rate on examples containing a relative clause was much higher at 16.16%. While the model struggles with relative clauses, it does capture the syntactic structure involving relative clauses as described in the next section.

4.4.1 LSTM Cell Activations

To understand the behavior of the model and what syntactic structure it is learning, we analyze the model's activation on a pair of constructed sentences similar to Linzen et al. The following are the constructed sentences utilized:

> The **houses** of the man from the office across the street ...
> The houses that the **man** from the office across the street ...

The first sentence has a prepositional phrase as indicated by the word "of", keeping the active subject of the sentence at "houses". However, the second



(a) (b)

Figure 3: Verb number prediction error rate depending on the number of homogeneous agreement attractors between the subject noun and the verb. Figure 3a shows verb number prediction error rate as compared to noun-only baselines (Section 3.2.1). Figure 3b shows verb number prediction error rate as compared to grammaticality baselines (Section 3.2.2).

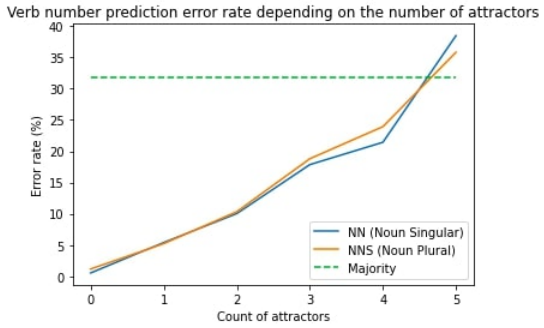


Figure 4: Verb number prediction error rate depending on the number of agreement attractors and the type of the subject noun

sentence has a relative clause indicated by the word "that", changing the active subject of the sentence to "man".

As observed in Figure 5, while the activations for both sentence begin with the probability that the verb is singular is close to 1, the activations diverge after the critical word "of/that". For the sentence with the prepositional phrase, the probability that the verb is singular drops close to 0 after the prepositional phrase starts. The model ignores the agreement attractor and continues to keep the active noun as "houses". However, for the sentence with the relative clause, the probability that the verb is singular remains close to 1 after the relative clause starts. The model sees the noun "man" after the relative clause starts and changes the focus of the active noun to "man".

The LSTM cell portrayed in Figure 5 is not necessarily how every LSTM cell in the model may have learned syntactic structure. The cell picked for demonstration purposes learned the structure very well, but many cells especially struggled with the relative clause structure, alluding to the inherent problem.

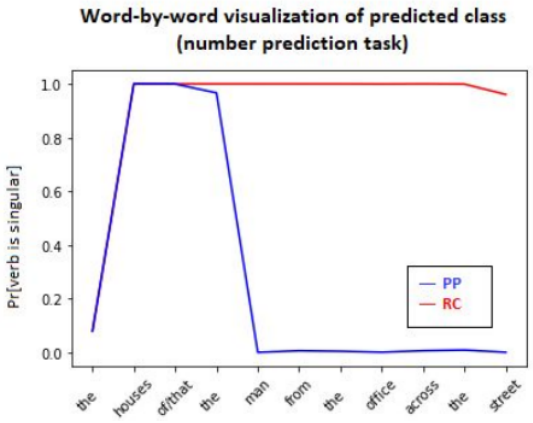


Figure 5: Word by word visualization of activation of a particular LSTM cell in the Number Prediction model for prepositional phrases and relative clauses.

4.5 Extension: Number Prediction on randomized subjects

For Number Prediction on randomized subjects, we evaluate our models on a modified dataset (as described in Section 3.4) and summarize our results in Table 2.

Unsurprisingly, we observe that our baseline model (trained with the unmodified dataset) performs poorly when evaluated on the modified

Model	Accuracy	F1-score
Baseline	98.81	98.14
Train on original Eval on random subj.	83.3	73.0
Train + Eval on random subj.	94.0	90.6
Train on random subj. Eval on original	95.1	92.6

Table 2: Accuracy and F1 score results for the verb number prediction task, evaluated on sentences with randomly sampled subjects. Baseline refers to our model trained and evaluated on the unmodified dataset.

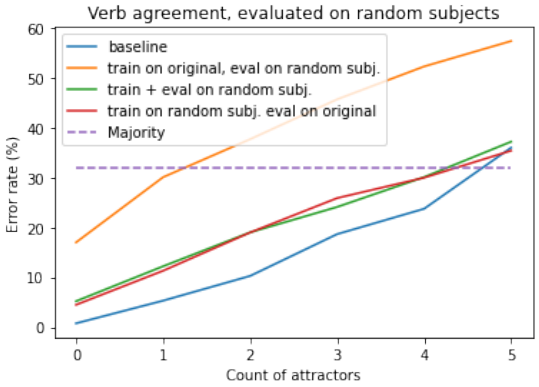


Figure 6: Verb number prediction error rate depending on the number of agreement attractors. Baseline refers to our model trained and evaluated on the unmodified dataset.

dataset. We measured an F1-score of 73%, which lies around than the majority label proportion of 68% (Table 2). When we analyze error rates on sentences with varying agreement attractor counts, we observe that our baseline model does acceptably when zero attractors are present, with an error rate that stays under the majority label (Figure 6). However, when evaluated on sentences with just one agreement attractor, the baseline model performs similarly to random guessing.

The poor performance of the baseline can be attributed to the model learning to memorize specific co-occurring words when training with the unmodified dataset. Co-location and context matters for natural language tasks, and learning these feature weights is expected and effective for the number prediction task.

Our model that is trained on the modified dataset scored relatively well. We measured an F1-score of 90.6%, which is slightly lower than the baseline. This model did relatively well on sentences with higher attractor counts, and we observe that error rates increase over agreement attractor counts at a rate similar to that of the baseline (Figure 6). We also evaluate this model on the unmodified dataset to ensure that it did not learn any co-locational information, and we find that the model performs similarly on both datasets (Table 2, Figure 6).

The overall accuracy and F1-score achieved by this model appears similar to that of the nouns-only baselines. We observe that this model performs slightly worse in the case of zero attractors, but does not experience the same sharp increase in error rate with higher agreement attractor counts. We believe that the difference between our model trained on the modified dataset and the baseline can be attributed to a lack of contextual information, similar to that of the nouns-only baselines.

We conclude that LSTMs are capable of capturing language structure in the absence of co-locational information. This is similar to results achieved by Kuncoro et al. (2018).

4.6 Extension: Probing Number Prediction for POS tags

	Accuracy	F1-score
Overall	65.9	68.2
unweighted avg across all classes	68.4	64.1

Table 3: Accuracy and F1 score results for POS probing on a model trained on the number prediction task.

Predicted	Gold	% of mistakes
NN	JJ	6.7
NN	NNP	5
IN	NN	2.3
JJ	NN	2.3
DT	NN	2.1

Table 4: Five most frequent mistakes in POS probing. POS tags from Penn Tree Bank.

For probing the Number Prediction task, we take a model trained on the Number Prediction task and probe for POS tags as described in Section 3.5. We summarize overall accuracy and F1 scores in Table 3 and report per POS tag results in the Appendix (Table 5). We also list the five most frequent mistakes made by our model in Table 4.

Our model was able to successfully predict POS information, with overall and unweighted average

accuracy and F1 scores significantly higher than can be achieved through random guessing (2.2%) or predicting the most probable class (24.4%). This suggests that our LSTM model trained on Verb Number prediction captured POS information in its latent representation, without explicit POS supervision. This result is expected: a model that performs well on subject-verb agreement must implicitly learn some form of syntactic structure.

Most mistakes made in the probing task relate to false positives and false negatives of NN, NNS, VBZ, and VBN, as shown by Table 4. Subject-verb agreement only strictly requires that our model learns to differentiate between singular and plural, subjects and verbs, and potentially determiners. Surprisingly, our model learned to differentiate between other POS tags relatively well (Table 5). This suggests that models trained with explicit supervision for specific language structures can implicitly learn other syntactic structures.

4.7 Language Model Probing

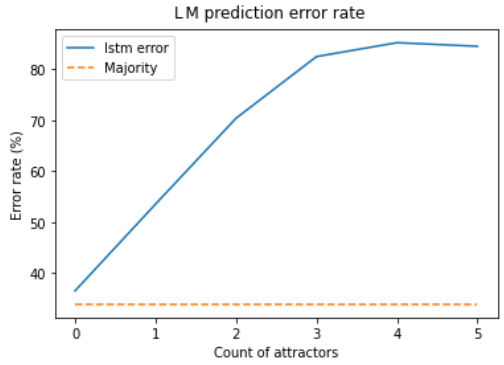


Figure 7: Probing subject-verb agreement error rate depending on the number of agreement attractors.

For probing the LM task, we train a model on the LM task and probe for subject-verb agreement as described in Section 3.3. Probe evaluation results are summarized in Figure 7.

We measure significantly high error rates on the probing task, even for sentences with zero attractors between the subject and verb. Sentences with one or more attractors achieve error rates much higher than the majority (32%) and random guessing (50%).

Our accuracy (60%) is much lower than that reported by Linzen et al. (92%). We believe this might be due to our change in methodology (as mentioned in Section 3.3, we were only able to train and evaluate on ~25% of the dataset due to limited compute resources).

5 Conclusion

Overall, we were able to reproduce almost all results and analysis from the Linzen et al. paper. The error rates for the majority of the models were similar to those shown by Linzen et Al. We observed that the LSTM models that were trained under direct supervision did understand syntactic context as demonstrated against the nouns-only baselines and the grammaticality baselines. Additionally, while our models struggled with sentences containing relative clauses, they did capture the syntactic structure of the sentence as demonstrated by the cell activations and POS probing results.

We also show that models trained under direct supervision successfully learned subject-verb agreement, even in the absence of some co-locational information (e.g. co-occurring nouns and phrases).

Finally, we show that a model trained for the subject-verb agreement task implicitly captures other sentence structure, specifically POS tags. Although results from self-supervised learning (without any labels for syntactic structure) are poor

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

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

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

Sepp Hochreiter. 1998. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6(02):107–116.

Rafal Józefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *CoRR*, abs/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. pages 1426–1436.

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.

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

A Appendices

POS tag	Precision	F1 score	% of dataset
NN	56.64	65.24	0.216
IN	63.47	69.88	0.153
DT	76.03	80.23	0.132
NNS	67.76	74.38	0.074
JJ	37.36	33.00	0.062
NNP	61.78	50.74	0.048
VBZ	65.94	65.55	0.046
CC	79.07	84.39	0.033
RB	41.98	35.56	0.028
VBP	62.35	59.05	0.022
TO	60.06	66.76	0.015
VB	52.90	37.63	0.011
PRP	63.96	54.48	0.010
VBN	35.77	17.90	0.007
VBG	62.63	35.42	0.007
WDT	74.63	74.13	0.006
POS	90.37	84.23	0.004
PRP$	47.74	27.61	0.003
WRB	72.02	77.03	0.003
MD	51.48	37.01	0.003
CD	43.30	12.61	0.003
WP	80.74	83.27	0.002
RBR	52.89	42.47	0.001

Table 5: Accuracy and F1 score results for POS probingon a model trained on the number prediction task, by POS tag.

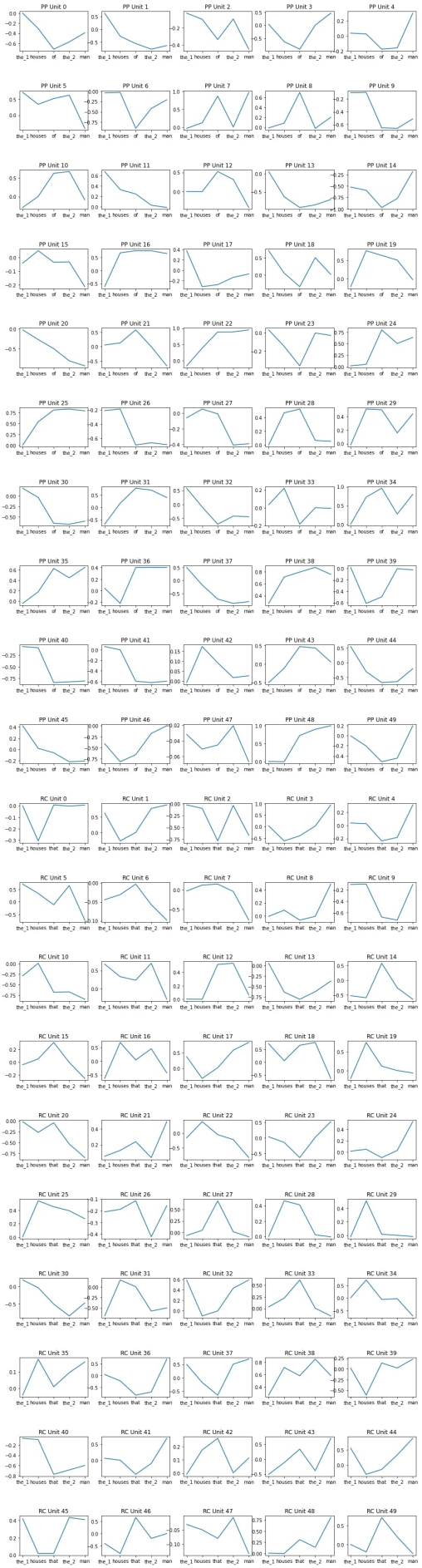


Figure 8: All the Prepositional Phrase LSTM units and the Relative Clause LSTM units