Recurrent Neural Networks (RNNs) performs well on SVA (Subject Verb Agreement)

VEERBHADRA M BHARATI
vb1779@gmail.com

Asst. Prof. Rakesh Patil
rakesh.rl@gmail.com
Tilak Maharashtra Vidyapeth (TMV) Department of Computer Science
Pune , MH - 411037.

Abstract

Recurrent Neural Networks (RNN) use sequential data to solve common temporal problems seen inlanguage translation and speech recognition. RNN are well know for their memory as they take information from prior instances / inputs to impact their current input and output. While traditional DNR (Deep Neural Networks) takes into assumption that inputs and outputs are autonomous to each other, the performance of RNN depends on the previous elements with the sequence. While future events would be supportive in identifying the output of a given sequence, unidirectional recurrent neural networks cannot account forthese events in their predictions. Our observations suggest that RNNs being fundamentally statistical models can efficiently capture the correlation of the output variable with the input as observed during training, even for relatively hard or nonlinear linguistic dependencies like Subject Verb Agreement (SVA),without necessarily learning the underlying hierarchical structure. Keywords: Analysis, Investigation, Research, LSTM, RNN, AI, SVA, Pattern recognition, Computer Vision.

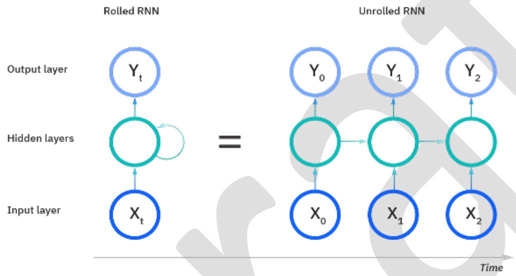
Keywords: Analysis, Investigation, Research, LSTM, RNN, AI, SVA, Pattern recognition, Computer Vision

Introduction

Acoustic modeling with DNNs and RNNs has commonly used the hybrid approach [1], where the neuralnetworks as discriminative models estimate the posterior probabilities of phonetic states most commonlyhidden Markov model (HMM) states. Let’s take an idiom, such as feelings of the regional weather, which is usually used when we feel ill, to help us in explanation of RNNs.

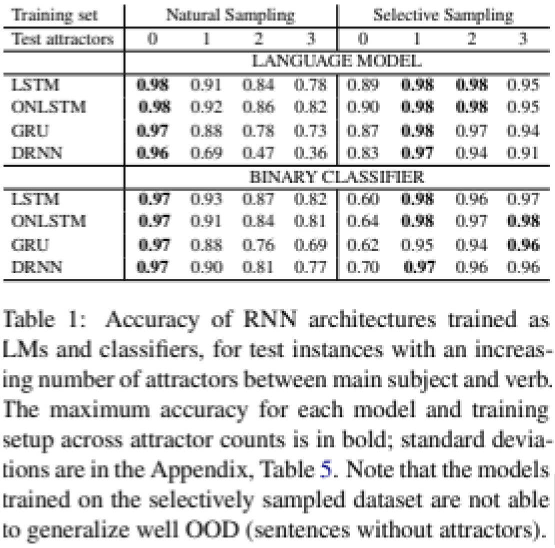
In order for the idiom to make sense, it needs to be expressed in that specific order. As a result, recurrent networks need to account for the position of each word in the idiom and they use that information to predict the next word in the sequence. Looking at the visual below, the rolled visual of the Recurrent Neural Networks (RNN)denotes the entire neural network, or rather the every foreseen phrase, like feeling under the weather. Another unique characteristic of RNN is that they share constraints across respective layer of the network.

While feed forward networks have diverse weights through every node, RNN share the same weight factor inside every layers of the network. As said, these weights are still accommodate in the through the processes of back propagation and gradient descent to facilitate RNN understands. Recurrent Neural Networks uses back propagation over time (BPTT) algorithms to identify the gradients, which is somewhataltered from traditional back propagation as it is exact to ordered data. The unrolled visual represents the specific layers, or time steps, of the NNs. Every layers map to an individual words inthat phrase, such as weather. Previous inputs, such as feeling and under, would be denoted as a hiddenstate in the 3rdstep to predict the outcome in the sequence of words. The codes of BPTT are the same as old-fashioned back propagation, where the model trains itself by calculating errors from its output layers to its input layers. These calculations permits us to alter and accommodate the parameters of the model suitably. BPTT varies from the old-fashioned approach in that BPTT sums errors at each time steps while feed forward networks don't need to sum errors as they do not share parameters across each and every layers. Through these processes, RNN tend to run into 2hitches, known as discharge gradients and is appearing gradients. These issues are different by the size of the gradient, which is the slope of the loss function along the fault curve. When the gradient is too small, it lasts to become smaller, updating the weight parameters until they are irrelevant. When this happens, the algorithm is not learning any more. Exploding gradients occur when the gradient is too huge, making an unsteady model. In that case, the model weights will grow too huge, and they tend to finally be represented as NaN. One answer to these kind of issues is to lessen the number of hidden layers within the neural network, removing some of the difficulty in the RNN models that are available.



Methodology

Our observations suggest that RNNs being fundamentally statistical models can efficiently capture the correlation of the output variable with the feedback as detected during training, even for relatively hard or nonlinear linguistic dependencies, without necessarily learning the underlying hierarchical structure. This is consistent with the conclusions of Sennhauser and Berwick (2018) and Chaves (2020). Thus, we tend to be careful in deducing the capability of such models to apprehend syntax sensitiveness needs. Performance on any exact kind of construction might always return some over fitting to it, even if it is syntactically rich or complex. Broadbased testing on instances of diverse types and complexity levels is essential to the development of models which better capture the structure of human language in all its richness and variety. In these kind of work, we tend to draw attention on assessing the models capability to make grammaticality findings when qualified for classification (supervised) and language modeling (self-supervised). For every tasks, we train models (withfive random seeds) on both training subsets from the corpus. Study the sentences from the introduction. A classifier is likely to label sentence 1 as un-grammatical and Sentence 2 as grammatical.



Training set	Natural Sampling				Selective Sampling			
Test attractors	0	1	2	3	0	1	2	3
	LANGUAGE MODEL							
LSTM	**0.98**	0.91	0.84	0.78	0.89	**0.98**	**0.98**	0.95
ONLSTM	**0.98**	0.92	0.86	0.82	0.90	**0.98**	**0.98**	0.95
GRU	**0.97**	0.88	0.78	0.73	0.87	**0.98**	0.97	0.94
DRNN	**0.96**	0.69	0.47	0.36	0.83	**0.97**	0.94	0.91
	BINARY CLASSIFIER							
LSTM	**0.97**	0.93	0.87	0.82	0.60	**0.98**	0.96	0.97
ONLSTM	**0.97**	0.91	0.84	0.81	0.64	**0.98**	0.97	**0.98**
GRU	**0.97**	0.88	0.76	0.69	0.62	0.95	0.94	**0.96**
DRNN	**0.97**	0.90	0.81	0.77	0.70	**0.97**	0.96	0.96

Table 1: Accuracy of RNN architectures trained as LMs and classifiers, for test instances with an increasing number of attractors between main subject and verb. The maximum accuracy for each model and training setup across attractor counts is in bold; standard deviations are in the Appendix, Table 5. Note that the models trained on the selectively sampled dataset are not able to generalize well OOD (sentences without attractors).

1. REVIEW OF RELATED STUDIES

Previous work (Linzen et al., 2016; Marvin and Linzen, 2018; McCoy et al., 2018; Kuncoro et al., 2019;Noji and Takamura, 2020; Hao, 2020) assessed the ability of RNN Language Models (LMs) to capture syntax sensitive dependencies. However, it is still not clear if good performance on SVA tasks is necessarily a result of the RNNs ability to capture the underlying syntax, and this is the question we seek tofurther investigate here. On the other hand, Chaves(2020) and Sennhauser and Berwick (2018) provide evidence that LSTM models are more likely to learn surface level heuristics, However, it is still not clear if good performance on SVA tasks is necessarily a result of the RNNs ability to capture the underlying syntax, and this is the question we seek to further investigate here. as well as the inputs, helps to generalize to unseen sentences. On the other hand, Chaves (2020) and Sennhauser and Berwick (2018) provide evidence that LSTM models are more likely to learn surface level heuristics, such as agreeing with the most recent noun, than the underlying grammar. We test the hypothesis that if the models under consideration were to capture the correct grammatical structure from syntactically rich input, then they would be able to generalize out of distribution (OOD), i.e. when tested on sentences without attractors having been trained solely on sentences with at least one In our experiments, we compare this setting to the more natural one of models trained on a dataset without any restriction on the number of attractors. Purpose of the research is to identify out that despite providing robust hierarchical cues via a selectively sampled training set, RNNs do not simplify to hidden combination. To find out that a soft hierarchical inductive bias, as communicated by the ONLSTM, in total to a syntactically rich training set, is also lacking to detect the basic grammar of NL as in case of SVA (Subject Verb Agreement). To catch that our results are steady across multiple learning models, self-supervised language modeling and supervised grammaticality ruling, as well as diverse test sets, natural and built.

2. HYPOTHESES OF THE STUDY

Responsiveness of Speech Recognition and Language Recognition/Translation. There are no noteworthychanges between Speech Recognition, Language Translation and SVA

Methodology And Analysis

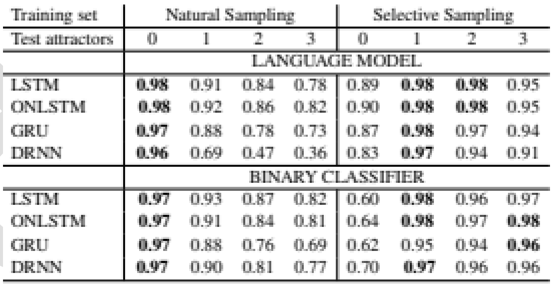
1. Population And Sample

We use sentences from the Wikipedia corpus made available by Linzen et al. (2016). For training, we took two subsets from the primary dataset, based on the number of attractors in each and every sentence (Figure 1). The sentences with no attractor are grammatically simple and allowed for out of distribution testing as are not viewed while training on the respective sampled dataset. For the binary classifier, we supplement each sentence with its respective counterfactual example.3 We just did testing on the sentences from the corpus (157k), we also tested our models on synthetically errors generated by sentences for specific syntactic evaluation (Marvin and Linzen, 2018).

2. Statistical Techniques Used in the Present Study

In this work, we conduct our experiments on four recurrent schemes LSTM (Hochreiter and Schmidhuber, 1997), GRU (Cho et al., 2014), Decay RNN (DRNN) (Bhatt et al., 2020), and ONLSTM (Shen et al 2019).The governing equations of these architectures are mentioned in A.1. ONLSTM is a recurrent network with soft hierarchical inductive bias.

3. Data Analysis and Interpretation



Training set	Natural Sampling				Selective Sampling			
Test attractors	0	1	2	3	0	1	2	3
	LANGUAGE MODEL							
LSTM	**0.98**	0.91	0.84	0.78	0.89	**0.98**	**0.98**	0.95
ONLSTM	**0.98**	0.92	0.86	0.82	0.90	**0.98**	**0.98**	0.95
GRU	**0.97**	0.88	0.78	0.73	0.87	**0.98**	0.97	0.94
DRNN	**0.96**	0.69	0.47	0.36	0.83	**0.97**	0.94	0.91
	BINARY CLASSIFIER							
LSTM	**0.97**	0.93	0.87	0.82	0.60	**0.98**	0.96	0.97
ONLSTM	**0.97**	0.91	0.84	0.81	0.64	**0.98**	0.97	**0.98**
GRU	**0.97**	0.88	0.76	0.69	0.62	0.95	0.94	**0.96**
DRNN	**0.97**	0.90	0.81	0.77	0.70	**0.97**	0.96	0.96

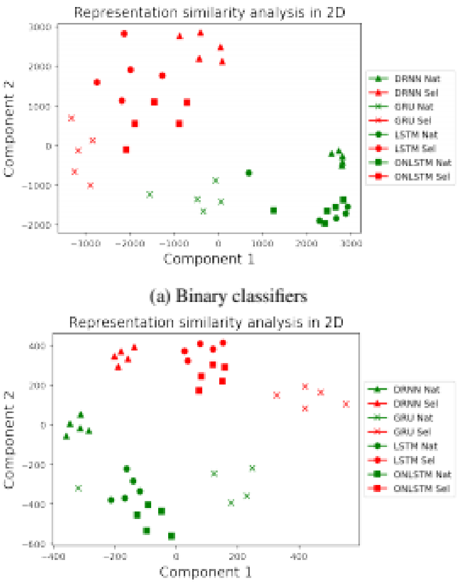
Table 1: Accuracy of RNN architectures trained as LMs and classifiers, for test instances with an increasing number of attractors between main subject and verb. The maximum accuracy for each model and training setup across attractor counts is in bold; standard deviations are in the Appendix, Table 5. Note that the models trained on the selectively sampled dataset are not able to generalize well OOD (sentences without attractors).

Performance on Natural Sentences

Table 1 shows the main results for the described experiments. For the models trained on a naturally sampled dataset, the performance reduces faster with an cumulative number of attractors between the subject & the respective verb, for both the LM & classifiers version. However, the decrease in the correctness with growing attractor count for the models trained on the chosen sampled dataset is fewer than with the natural sampling training. For the selected sample dataset, the sentences with no attractors help as OOD sentences, and performance improvement on distribution difficult sentences comes at the price of a decrease in the accuracy on the OOD for moderately simple sentences. The error rate for the ONLSTM, a model withinherent tree bias, also surges when tested on the OOD sentences, &when practiced on a classification objective it attains no good result than the architecturally simpler Decay RNN. This falloff on grammatically simpler OOD samples seems counterintuitive. We note that the surge in error rates is much larger when we train the models as classifiers rather than the LMS. This indicates that models with supervised training for grammaticality on syntactically rich and counterfactually augmented data we still are inept to capture the real syntactic rules, &appear to be learning shallower heuristics, but ones which capture more nuanced patterns than simply going by linear distance. We can deduce this because while our selected sampled subset contains sentences with at 1 attractor, more been (over 30%) of the dominant nouns in these sentences are non-attractors. Hence the sentences in which a non-attractor noun (same number as the main subject) instantly leads the verb somewhat than an attractor noun. Therefore the agreement performance (sentences with attractors) of the models qualified on this dataset cannot ascend from an excessively simple heuristic like conflicting with the most recent noun, and the detected decline in OOD performance suggests that fewer unimportant heuristics are being learned which yet fail to capture the real syntax.

1. Analysis of representations:

To study the variances in the learned internal representations between the models trained on the 2 subsets of the data, we achieve a representation resemblance analysis (RSA) (Laakso and Cottrell, 2000). Wetake 2000 sentences chosen randomly from the test set. Our foremost reflection from Figure 2 is that the illustrations of models proficient on diverse subsets are effortlessly linearly divisible in this space, for both the LM and the classifier objectives. This advises that the representation clustering is not so much based on model architecture or inductive bias, but are focused more by the training data.



(a) Binary classifiers

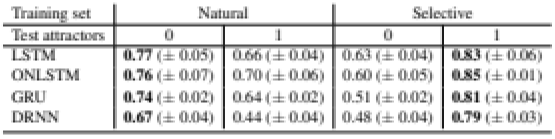
(b) Language models

Figure 2: Representation similarity analysis of the hidden units of different RNN models (5 different seeds for each model). We observe that for both the learning objectives, one can partition the 2D space using a line which separates models trained on the two subsets of the data, natural and selective sampling.

To evaluate the performance of the models trained on the chosen sampled dataset, we need to take a closer look at built sentences that are structurally alike to the distribution sentences but contain non-attractor prevailing nouns somewhat than the agreement attractors. Figure 3 indicates the performance of the LSTM LM on 3 agreement environments across Object RC, Preposition Phrase, and Subject RC, respectively with animate main noun. We observe that with our selective training, the performance on sentences with non-attractor intervening nouns (the SS/PP configurations, which are unobserved in the selectively sampled dataset) worsens substantively for 2 out of3 syntactic constructions across Preposition Phrase and Subject RC.

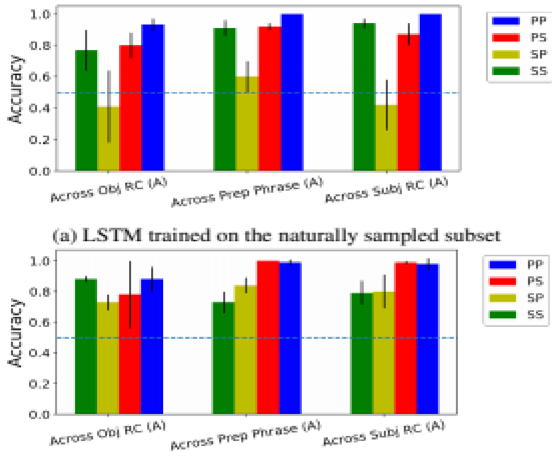
Result And Discussion

In this work, we examined the properties of a purpose fully selected training set with completely hard agreement instances, on NL models and binary classifiers for grammaticality decisions. We detected that the models incapability to achieve well on out of distribution (OOD) sentences, even those which may seem to be easy agreement instances, is reliable across difference in learning mechanism (supervised or self-supervised), distinctive architectural bias, & testing set natural or artificial sentences. Our examination showed that fault rates of models trained on sentences with at least 1 agreement attractor are higher on sentences with no attractors than on sentences with attractors, for both corpus sentences (Table 1) and artificial sentences (Table 2). This remark is counterintuitive because the representations were trained on syntactically rich sentences and failed to achieve well on SVA. Had our RNN models selected up the right grammatical rules as in SVA, we won’t suppose this behavior. We obtained a similar counterintuitive outcome for targeted syntactic evaluation (Appendix, Table 6), where models are trained on the selectively sampled dataset achieved much improved on tough constructed sentences involving agreement across nested dependencies, rather than simpler sentences connecting agreement inside the nested dependencies.



Training set	Natural		Selective	
Test attractors	0	1	0	1
LSTM	**0.77** (± 0.05)	0.66 (± 0.04)	0.63 (± 0.04)	**0.83** (± 0.06)
ONLSTM	**0.76** (± 0.07)	0.70 (± 0.06)	0.60 (± 0.05)	**0.85** (± 0.01)
GRU	**0.74** (± 0.02)	0.64 (± 0.02)	0.51 (± 0.02)	**0.81** (± 0.04)
DRNN	**0.67** (± 0.04)	0.44 (± 0.04)	0.48 (± 0.04)	**0.79** (± 0.03)

Table 2: Accuracy of LMs on test instances with 0 or 1 attractors from the artificial corpus. Models trained on the selectively sampled subset do not generalize well on OOD sentences without attractors. Performance across different syntactic constructions is shown in Table 6 in the Appendix.



(a) LSTM trained on the naturally sampled subset

(b) LSTM trained on the selectively sampled subset

Figure 3: Fine-grained analysis of the LSTM LM on Obj/Subj Relative Clauses and Preposition Phrases, demarcated by the inflections of the main subject and the embedded subject. P: Plural, S: Singular; thus SS denotes sentences with a singular main noun and a singular embedded subject, and likewise for the other cases.

Conclusion

- Our analysis of representations suggested that training set bias dominates over the models architectural features or inductive bias in shaping representation learning; e.g., there was no discernible difference between the learned representations of ONLSTM and LSTM models.

- The reasons for this merit further exploration. Moreover, for the binary classifiers (Figure 2a), although we observe little variance in test accuracy across different training seeds, the variance in the projected representation space is substantially greater than for LMs. We suggest that an LM objective is more reliable when equating the ability of diverse RNN models to capture syntax delicate dependencies.

- We experimented that the hierarchical inductive bias in the ONLSTM is not adequate to achieve well on OOD sentences. McCoy et al. (2020) opposed that architecture with explicit tree bias, plus syntactically annotated inputs, are needed to capture syntax for sequence to sequence tasks.

- Here we show that the ONLSTM (soft tree bias) trained on a syntactically rich dataset (soft structural information) turns out to be inadequate to streamline well to OOD sentences and capture the underlying SVA. Our targeted syntactic evaluation pinpoints the cases which our models fail to capture, and improving performance on such cases is a key future direction.

- Our observations suggest that RNNs being fundamentally statistical models can efficiently capture the correlation of the output variable with the input as observed during training, even for relatively hard or nonlinear linguistic dependencies, without necessarily learning the underlying hierarchical structure. This is consistent with the conclusions of Sennhauser and Berwick (2018) and Chaves (2020).

- Thus, we need to be cautious in inferring the ability of such models to capture syntax sensitive dependencies.

- Performance on any particular kind of construction might always reflect some overfitting to it, even if it is syntactically rich or complex. Broadbased testing on instances of diverse types and complexity levels is essential to the development of models which better capture the structure of human language in all its richness and variety

References:

[1] Gantavya Bhatt, Hritik Bansal, Rishubh Singh, and Sumeet Agarwal. 2020. How much complexity does an RNN architecture need to learn syntax-sensitive dependencies? In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pages 244– 254, Online. Association for Computational Linguistics.

[2] J Kathryn Bock and Carol A Miller. 1991. Broken agreement. Cognitive psychology, 23(1):45–93.

[3] Rui Chaves. 2020. What don't RNN language models learn about filler-gap dependencies? In Proceedings of the Society for Computation in Linguistics 2020, pages 1–11, New York, New York. Association for Computational Linguistics.

[4] Tilak, G., 2021. Robotics and Artificial Intelligence: Impact of Robotics at Intelligent Homes. International Journal of Applied Engineering Research, 6(1), pp.11-30.

[5] Kyunghyun Cho, Bart van Merrienboer, Caglar Gul- ¨cehre, DzmitryBahdanau, FethiBougares, Holger Schwenk, and YoshuaBengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In Proceedings of the 2014

[6] H. Su, G. Li, D. Yu, and F. Seide, "Error back propagation for sequence training of context-dependent deep networks for conversational speech transcription," in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2013, pp. 6664–6668.

[7] K. Vesely, A. Ghoshal, L. Burget, and D. Povey, "Sequence- ´ discriminative training of deep neural networks," in INTERSPEECH, 2013.

[8] G. Heigold, "A log-linear discriminative modeling framework for speech recognition," Ph.D. dissertation, RWTH Aachen University, Aachen, Germany, Jun. 2010.

[9] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," Signal Processing, IEEE Transactions on, vol. 45, no. 11, pp. 2673–2681, 1997.

[10] F. Eyben, M. Wollmer, B. Schuller, and A. Graves, "From speech to letters using a novel neural network architecture for grapheme based ASR," in Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on. IEEE, 2009, pp. 376–380

[11] [Foroutan and Sklansky, 1985] Foroutan, I. and Sklanskv, J. Feature Selection for Piecewise Linear Classifiers. In IEEE Proc. on Computer Vision and Pattern Recognition. San Franscisco, 1983,149-154.