Positive Bias and Length Bias: How Deep Neural Sequence Differs to Human Cognition in Premises Selection

Adam Huang (huanga2@carleton.edu)
Department of Computer Science, 300 North College Street
Northfield, MN 55057 USA

Terry Wang (wangy4@carleton.edu)
Department of Mathematics, 300 North College Street
Northfield, MN 55057 USA

Abstract

We compare here the premises selections of the deep neural sequence models with that of human cognition. We train fully-connected neural networks with different numbers of hidden units and an LSTM using conjecture and axiom pairs from the Mizar Mathematical Library and compare the networks' performances. We then compare the premises choices for a sample of 5 conjecture and axiom pairs from 9 college students to that of our network model. We find that the network models tend to classify more premises as necessary and predict more accurately with greater number of premises, which contrasts with the pattern humans exhibits. We propose that the Positive Bias and the Length Bias makes the network models distinct from human cognition.

Keywords: Deep Learning, Premises Selection, Automated Theorem Proving

Introduction

Overview

Mathematical reasoning is unique to human cognition and intelligence. With mathematics, human beings can generalize abstract knowledge from concrete examples and use the knowledge to make useful inferences. Understanding mathematical reasoning, therefore, lies at the heart of understanding human cognition (Lakoff & Núñez,2000).

As mathematical reasoning is the prerequisite for theorems proving, investigating the process of proving theorems can inform us how mathematical reasoning is formalized in human brain. Since the 1950s, several attempts have been made to develop models that generate human proofs. Unfortunately, these models have their limitations, with one of the most severe issues being the lack of strong automated reasoning methods to fill in the gaps in already formalized human-written proofs (Irving et al.,2016). As a result, in a large corpus of computer-understandable reasoning knowledge, the current Automated Theorem Prover (ATP) performs poorly in the task of *premises selection* which refers to the selection of a set of axioms for proving a given conjecture.

Due to its success in computer vision and natural language processing, deep neural sequence models became a reasonable next choice for improving premises selection. Recent studies (Irving et al.,2016) and (Loos, Irving, Szegedy, & Kaliszyk,n.d.), which explore the performance of ATP systems in premises selection with the guidance of deep neural sequence models, have shown great success in the ability of deep learning to improve the overall performance of ATP systems.

In this work, we construct various types of deep neural networks to model premises selection in human cognition. Rather than only comparing across the model in terms of test accuracy and learning efficiency, we also compare the accuracy of the models in selecting necessary premises with the selections of 10 college students on 5 premises. The main discoveries of this work are the following:

- We find that, among all the models we investigate, the 1024×1024 fully-connected model is the best model for premises selection in terms of test accuracy and validation efficiency.
- We discover that the network models tend to classify more premises as necessary than unnecessary. We refer to this as the 'Positive Bias' of the network models.
- We also discover that the network models tend to perform better in proving conjectures that paired

with more axioms, which we call 'Length Bias'.

- Both the 'Length Bias' and the 'Positive Bias' of the models contrast with the human data.

Automated Theorem Proving

Automated reasoning refers to the study of understanding various types of reasoning using modeling techniques. One of the sub-fields in automated reasoning is Automated Theorem Proving (ATP), which focuses on formalizing verifiable theorems, definitions, relations, and proofs.

ATP systems are computer-program-based systems that facilitate theorem proving from translating and formalizing logic to writing proofs. The first ATP system dated back to Martin Davis's implementation of the Presburger's algorithm on JOHNNIACC in 1954 (Bibel,2007). In the new century, with the help of machine learning (Bridge,2010), the performances of deductive ATPs improved dramatically. Current first-order-logic based ATP systems such as the Lean Theorem Prover (Moura, Kong, Avigad, Van Doorn, & Raumer,2015), Z_3 prover (De Moura & Bjørner,2008), and E prover (Schulz,2013), with the support of multi-core machines, can prove theorems with high accuracy and efficiency.

Premises Selection

However, in large corpora of data, ATP systems generally perform poorly in the task of premises selection, with their major problem being the lack of strong reasoning methods in selecting necessary premises to prove a conjecture. We will introduce here the formal definition of the premises selection task.

Definition. ***(Premise)*** *A Premise is a proposition antecedently supposed or proved as a basis of argument or inference. In automated reasoning, premises are specifically modeled as conjectures and axioms, where the axioms are premises that are proved and conjectures are the premises to be proved.*

Therefore, we define our task as :

Definition. ***(premises selection problem)*** *Given a large set of premises P, a new conjecture C, and an ATP system A with given resource limits, predict those premises from P that will most likely lead to an automatically constructed proof of C by A.*

Mizar Mathematical Library

Our training data for premises selection are collected from the *Mizar Mathematical Library*. The library writes mathematical problems and theorems in first-order logic using context-free grammars. Specifically, our premises are collected from the proofs provided by the evaluation of the AI-ATP methods on version 4.181.1147 of the library from (Kaliszyk & Urban,2015).

We obtained a data set of 32,542 unique conjectures and 69,918 unique axioms. As one axiom can be used to prove several conjectures, we obtain a total number of 522,528 conjecture and axiom pairs. These conjecture and axiom pairs are labeled with '1' if they are necessary to prove the conjecture, and '0' if they are not. The labels are provided by the experiment from (Irving et al.,2016), with 261,727 necessary conjecture and axiom pairs and 260,801 unnecessary conjecture and axiom pairs.

The DeepMath Project and Our Motivation

Our work is motivated by the DeepMath project (Irving et al.,2016) conducted by Google in 2016. The project identifies the premises selection problem in ATP system and for the first time applies deep learning to ATP systems in a large scale. Another related work that builds on the DeepMath Project is (Loos et al.,n.d.) which engineers the network models built by DeepMath Project into an actual ATP system and shows that the ATP system improves both its accuracy and efficiency in writing proofs with the guidance of neural networks.

Although both works show that deep neural sequence models are helpful for premises selection, they only focus on quantitative comparison based on accuracy and efficiency between the models. As deep neural sequence models are motivated by actual architecture of human brains (Fodor & Pylyshyn,1988), it would also be useful to investigate whether the ATP systems, with the guidance of the networks, perform premises selection more closely to human cognition.

Therefore, we will compare the network models with human cognition. We first implement an LSTM and several fully-connected deep neural sequences with various numbers of hidden units. We then compare across the models, not only in terms of accuracy and efficiency, but also based on how well the models capture the premises selection in human brains. The detailed architectures of the models will be discussed below.

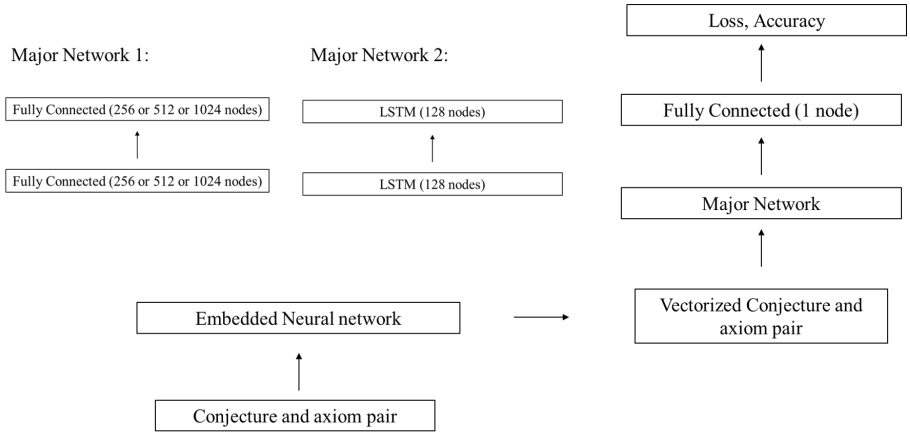

Figure 1: Our network architecture. The combined network is composed of an embedded network (Bottom Left) which vectorizes our conjecture and axiom pairs, and a major network (Right) for learning the categories. We use two types of deep neural sequence models for the major network: Fully-connected deep network and LSTM (Top Left).

Neural Network Architectures

Overview

The architecture of our combined network is summarized in Figure 1.The combined network can be divided into two parts: the embedded network (autoencoder) that vectorizes the conjectures and axioms pairs from Mizar40 (Kaliszyk & Urban,2015) into a tensor with form 1×512, and the major network that accomplishes premises selection.

We consider two architectures for the major networks: fully connected deep neural sequence and long-term short memory (LSTM).

Fully connected deep neural sequences

The fully connected deep neural sequences are composed of two layers. The number of nodes in each layer is equal, and can be either 256, 512, or 1024. We choose a rectified linear unit function as the activation function for both layers (Hummel & Biederman,1992).

LSTM

The second choice for the major network is a long-short term memory (LSTM) with two layers, where each layer is composed of 128 LSTM units. The LSTM layer remembers values over arbitrary time intervals and models as the long-term memory, and the three gates regulate the flow of information into and out of the cell. Compared to other recurrent neural networks, the LSTM tends to perform more accurately, learns faster, and solves both more complex and long-time lag tasks (Hochreiter & Schmidhuber,1997).

Experiments

Experiment 1: Positive Bias of the Deep Neural Network Models

Method We implement the network architectures above using the Keras interface and train the networks with 522,528 conjecture and axiom pairs from Mizar. We parse the pairs using the 'Lark Parser' into Function objects to make them readable by our embedded network, as shown in Figure 1. We split the data set into 90% training set and 10% test set, and make sure that the amount of necessary and non-necessary conjecture and axiom pairs in the test set are about equal. We also randomly pick 10% of our training data as our validation set during training. We set the loss function to be 'binary entropy' and minimize it using 'Adam Optimizer' and train each network architecture for 1500 epochs with batch sizes 4096. Finally, we compare the test accuracy and efficiency after training, which will be discussed in further detail below.

Test Accuracy and Validation Efficiency To compare across the models, we compare two quantities: test accuracy and validation efficiency. Given the experimental setting, we define accuracy as the proportion of the test conjecture and axiom pairs the models

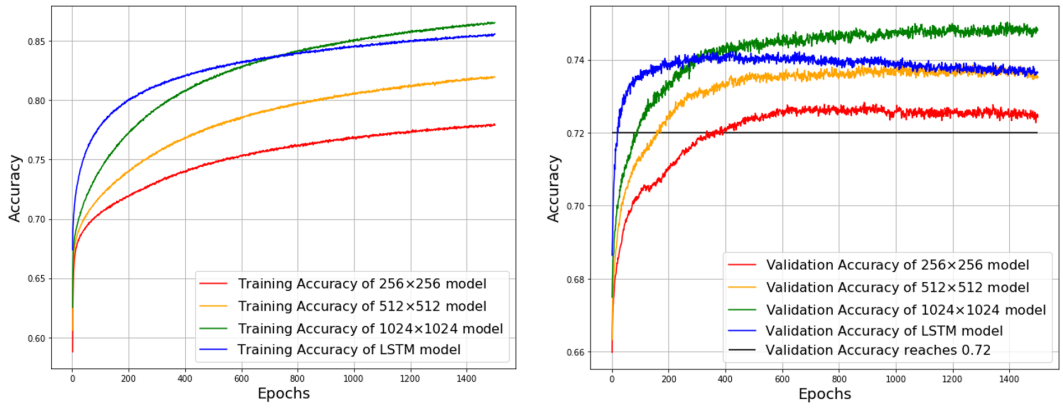

Figure 2: Computer simulation results. The left plot shows how training accuracy of the three fully-connected models and the LSTM change with epochs trained. The right plot shows how validation accuracy changes with epochs. A horizontal line is added at x = 0.72 to aid visualization of the validation efficiency.

categorizes correctly, and test accuracy as the accuracy of our test set. We define validation efficiency as the least number of epochs of training the model requires to reach a validation accuracy greater than 72 %. The lower the epoch number needed, the higher the efficiency.

Result Results for Experiment 1 are summarized in Figure. 2, Table 1, and Table 2.

Table 1: Table that records test accuracy and validation efficiency for each model.

Model	Test Accuracy	Valid' Efficiency
256 × 256	73.09	238
512 × 512	73.71	137
1024 × 1024	74.80	72
LSTM	73.80	17

Table 2: Table that records the false negative and false positive percentile of the test set.

Model	False Positive	False Negative
256 × 256	15.77	11.13
512 × 512	15.78	10.5
1024 × 1024	16.1	9.11
LSTM	14.03	12.14

Discussion Our results suggest that the 1024 × 1024 model works the best among the four models in premises selection. Additionally, we discover a phenomena called 'Positive Bias' of the models.

According to Fig. 2 and Table 1, the LSTM network, though it achieves a high training accuracy and high efficiency, performs badly in both the validation and the test set. Moreover, as loss decreases, the validation accuracy follows. This evidence suggest that the LSTM, with high variances, over-fits the training data.

Moreover, with both low training accuracy and low test accuracy (see Figure 2 and Table 1), the 256 × 256 fully-connected model could be considered a high-bias model. The 512 × 512 model, though it performs much better than the 256 × 256 model, has a much lower efficiency, and much lower accuracy in the test set, validation set, and training set than the 1024 × 1024 model.

Therefore, if only comparing across the models through validation efficiency and test accuracy, the 1024 × 1024 model would be the best model for premises selection. Future experiments are encouraged to explore the maximum number of nodes that could increase the test accuracy without over-fitting the data.

By looking at the error analysis results (see Table 2), the false positives outnumber the false negatives for all models, implying that the models make more mistakes in predicting non-necessary premises as necessary than the opposite. As the number of necessary and non-necessary conjecture and axiom pairs are about the same in our test set, the pattern also implies that the models tend to classify more pairs as necessary than non-necessary. We define this phenomena as the 'Positive Bias' of the network models.

We propose that a 'Positive Bias' would not be as harmful as the opposite 'Negative Bias' to the ATP systems: as the ATP systems are often implemented on powerful computers, if only a moderate amount of non-necessary examples are presented during the proof search, the ATP systems could still find the proof; on the other hand, if our networks have 'Negative Bias', the ATP systems may lose necessary information for the proof, and as a result may not eventually find a proof for the given conjecture. Thus, the 'True-Bias' nature of the network supports the conclusion of the experiment conducted by (Loos et al.,n.d.), namely, that deep neural sequences serve as useful guidance for proof searches.

As a healthy 'Positive Bias' would be preferred, the 1024×1024 model, with the highest true positives percentile (see Table 2), is still considered to be the best model.

Experiment 2: Comparison between the Models and Human Cognition

Participant Our participants are 9 students who have already taken multi-variable calculus. We are confident that our participants all understand mathematical logic and know how to select necessary premises when proving a theorem.

Method We provide our participants with a set of 5 conjectures and their corresponding necessary and unnecessary axioms. We ask the participants to select an axiom if they think it will be necessary premises to prove the conjecture. We use the same accuracy definition as in the previous experiment to compute the accuracy of human's choices. We then compare the humans' accuracy with the models'. We also look at specific conjecture and axiom pairs to explore whether there are systematic difference between how the network selects premises and how humans select premises.

Result Results are summarized in Figure 3, Table 3, and Table 4.

Discussion According to Table 3, all four models agree with humans for only 50% of the premises choices, demonstrating that there are systematic differences between premises selections of the models and that of the humans.

Moreover, results from Table 4 also suggest that humans have 'Negative Bias', as the false negative percentile of humans is greater than the false positive percentile. This contrasts with the models, as we have seen in the previous experiment that our models have 'Positive Bias'. We propose that a possible reason for humans having 'Negative Bias' is that humans only have a limited amount of time to process data. As a result, humans may skip steps in proving the conjectures, which makes them ignore some of the necessary axioms.

Last but not least, Figure 3 shows that humans perform better in premises selection when only a few axioms are provided for proving the conjecture than when an abundant amount of them are provided. However, the network models perform much better in premises selection where the conjectures are paired with many axioms. Further, we observe that the accuracy of the models increases with the amount of axioms paired with the given conjecture, implying a positive correlation between the two quantities. We called the correlation here the 'Length Bias' of the models, which we will discuss

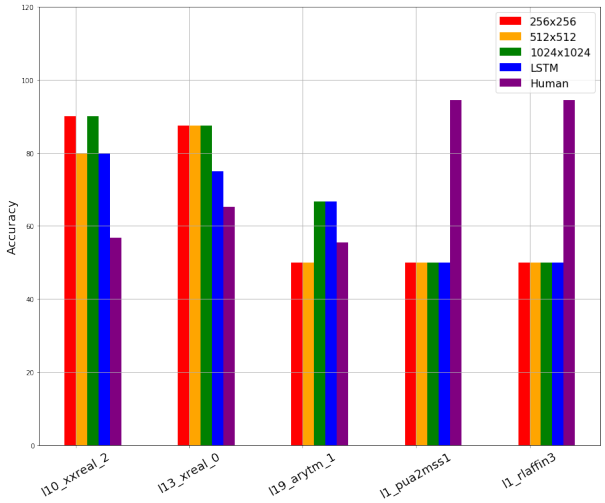

Figure 3: Comparison between humans' accuracy and neural networks' accuracy on the sample. The 5 conjectures in the chart are arranged in decreasing amount of axioms needed from left to right.

Table 3: Table that records the percentage of the premises humans agree with the model in the sample.

Model	Correct Percentile
256×256	0.45
512×512	0.48
1024×1024	0.46
LSTM	0.46

Table 4: Table that records the false negatives and false positives of humans on the 5 samples.

False Positive	False Negative
16.2	18.6

more formally in the next section.

Experiment 3: Investigation of the Length Bias

Definition of Length

We present here a formal definition of 'length' mentioned in the previous experiment.

Definition. ***(Length of a conjecture)*** *The length of a conjecture is the total number of conjecture and axiom pairs, both necessary and not necessary, for proving the conjecture.*

Method We retrain the networks with 469,448 training premises. The training premises are selected by subtracting the test set, which contains all axiom and conjecture pairs of 3200 test conjectures, from all premises. The process here prevents us from separating axioms from the same conjecture into two different sets, ensuring that the total number of conjecture and axiom pairs for all conjectures in both the test set and the training set equals the length of the conjecture.

Result

Results are summarized in Figure 4 , Figure 5, and Table 5.

Table 5: Records the correlation coefficients of test accuracy and length, for length between 0 and 25, for all four models.

Model	Corr' Coefficients
256×256	0.9872
512×512	0.9737
1024×1024	0.9854
LSTM	0.9616

Discussion

The results from the subplot in Figure 4 and Table 5 both suggest a strong positive linear correlation from length 1 to length 25 between average test accuracy and length of the conjectures for all network models, verifying the existence of the 'Length Bias' that we proposed.

However, as illustrated in the major plot in Figure 4, the average test accuracy stops increasing after the length reaches a threshold around 30. Further, we observe that the test average accuracy begins to decrease as the length approaches 80, implying that there might be a maximum accuracy the models can reach.

From Figure 5, we also find that, for all four models, the false negatives and false positives correlate differently with length. At first, both the amount of false positive and false negative decreases with increasing length, but after a certain threshold of length around 30, the amount of false positives begin to increase while the decreasing trend in the false negatives stay. As the turning point of the false positives agrees with where the bottleneck of test accuracy starts in Figure 4, such an increase may explain why the bottleneck would occur. We encourage future experiments to further explore the reason why such a bottleneck occurs.

General Discussion

(This section will be revised) The goal of this experiment is to investigate whether the premises selections of the deep neural models for guiding ATP proof searches are similar to that in human cognition. We discover that there are 'Length Bias': the network tends to make better choices in premises selection for conjectures paired with more axioms, and 'Positive Bias': the network favor classifying the premises as necessary than unnecessary, in the deep neural models, preventing them from predicting the premises human may select and the mistakes human could make.

The 'Length Bias' and 'Positive Bias' we points out here would be useful for ATP systems that combine the deep neural network techniques in proof search in the following way. First, as we have suggested, a healthy 'Positive Bias' will be helpful for ATP systems as the ATP would better searching the proof for a bit longer time rather than losing necessary information. However, as we have seen in the third experiment that the network reaches a bottleneck at some threshold of length due to the 'Positive Bias', so systematic ways to improve the 'Positive Bias' will be needed.

We acknowledge that the human data we have so far is not conclusive enough to illustrate the extent of difference between the deep neural network, so we encourage future experiments to take human data in a larger

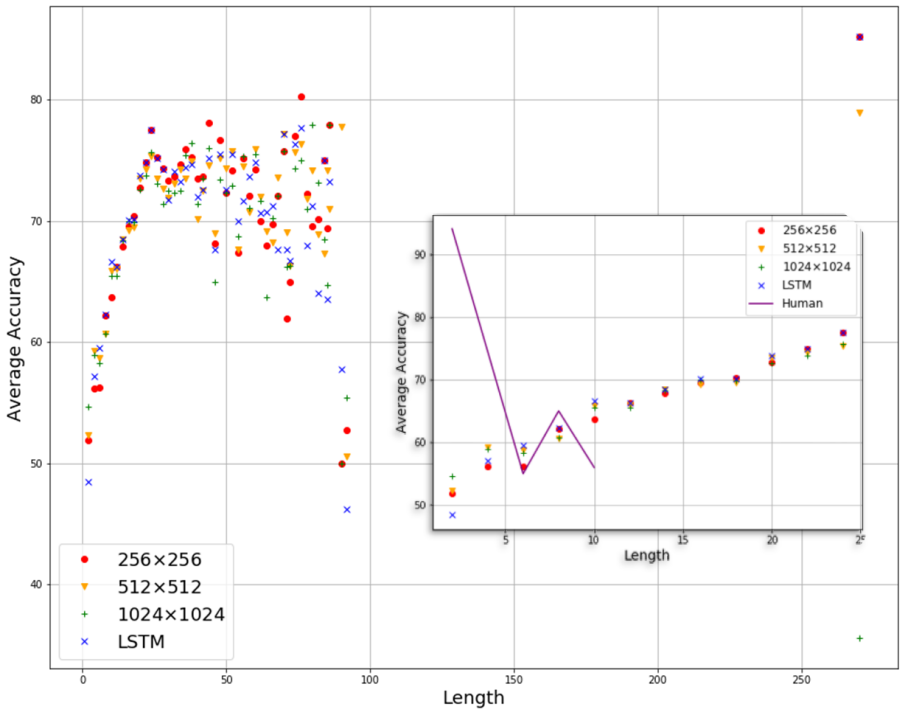

Figure 4: The plot of Average Test Accuracy versus Length of the Test conjectures. The major plot shows the whole data set, while the subplot exhibits a zoomed-in image of the plot from length 0 to 25. As the subplot demonstrates, average test accuracy scales linearly with length from length 0 to 25. The trend of the human subjects is also displayed in the subplot.

scale to verify our results. We also encourage future iterations to explore the reason for both the 'Positive Bias' and the 'Length Bias', we discover here.

Acknowledgments

Codes for parsing context free grammars and guidance for formatting and vectorizing the data are provided by Andrzej Kucik. Many thanks for his guidance! We also thanks our instructor of CS328 and advisor Anna Rafferty for all kinds of help she provides. Thanks to Zhaobin Li for his comments on editing the work. Thanks to Cameron Kline-Sharpe for editing this paper.

References

Bibel, W. (2007). Early history and perspectives of automated deduction. In *Annual conference on artificial intelligence* (pp. 2–18).

Bridge, J. P. (2010). *Machine learning and automated theorem proving* (Tech. Rep.). University of Cambridge, Computer Laboratory.

De Moura, L., & Bjørner, N. (2008). Z3: An efficient smt solver. In *International conference on tools and algorithms for the construction and analysis of systems* (pp. 337–340).

Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, *28*(1-2), 3–71.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Hummel, J. E., & Biederman, I. (1992). Dynamic binding in a neural network for shape recognition. *Psychological review*, *99*(3), 480.

Irving, G., Szegedy, C., Alemi, A. A., Eén, N., Chollet, F., & Urban, J. (2016). Deepmath-deep sequence models for premise selection. In *Advances in neural*

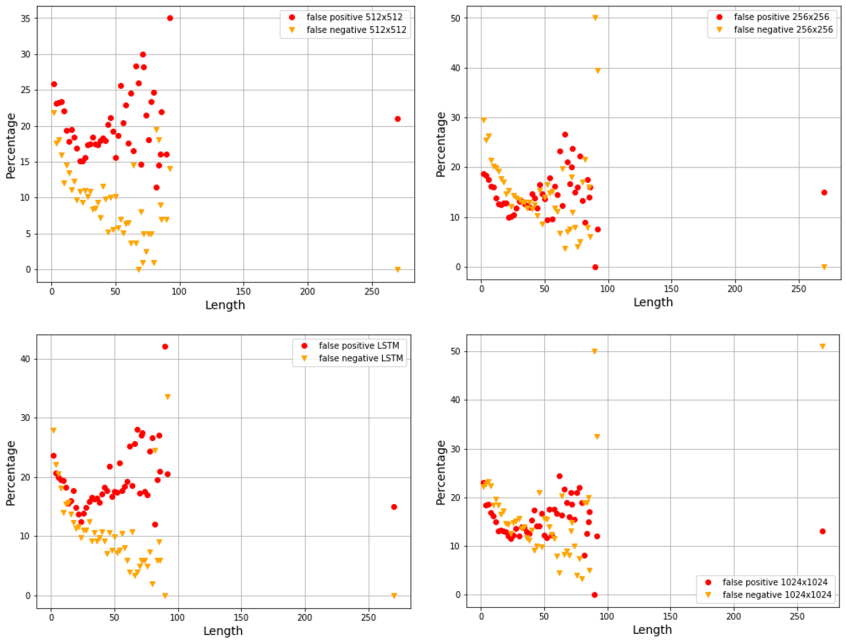

Figure 5: The plot of False Positive and False Negative percentile versus length for four neural networks. We see that after about length 40, the false positives turn from decreasing to increasing as length increases.

information processing systems (pp. 2235–2243).

Kaliszyk, C., & Urban, J. (2015). Mizar 40 for mizar 40. *Journal of Automated Reasoning*, *55*(3), 245–256.

Lakoff, G., & Núñez, R. E. (2000). Where mathematics comes from: How the embodied mind brings mathematics into being. *AMC*, *10*, 12.

Loos, S., Irving, G., Szegedy, C., & Kaliszyk, C. (n.d.). Deep network guided proof search.

Moura, L. de, Kong, S., Avigad, J., Van Doorn, F., & Raumer, J. von. (2015). The lean theorem prover (system description). In *International conference on automated deduction* (pp. 378–388).

Schulz, S. (2013). System description: E 1.8. In *International conference on logic for programming artificial intelligence and reasoning* (pp. 735–743).