Go For a Walk and Arrive at the Answer Track 2 - Executive Summary

Kayla Branson, McGill ID 260629913
Ashique Hossain, McGill ID 260572615
Samin Yeasar, McGill ID 260800927

I. INTRODUCTION

In this paper, we provide a summary of our findings from an ablation study [add footnote w/ full text] on the MINERVA model proposed by Das et. al in "Go for a Walk and Arrive at the Answer" [**?**]. MINERVA is a reinforcement learning-based model which answers queries on a knowledge graph. Given a knowledge graph of entities (nodes) and relations (edges), MINERVA learns to solve queries by finding unlabelled relations between entities.

II. SUMMARY

A. Reproduction of Results

In attempting to recreate the original paper's results, we found that MINERVA did not perform as strongly as reported. NELL-995 showed the most significant change, a decrease in MRR of 30% from 0.725 to 0.504. UMLS showed a similar performance to the original results, decreasing by around 1%, while other datasets showed between a 6-12% decrease.

B. Changing the Activation Functions

The original authors chose the ReLU activation function for the hidden layers of their MLP, without providing much justification for their choice. In order to evaluate their choice and the importance of ReLU units to the model, we tested a variety of alternatives, including cReLU, ELU, sigmoid and tanh functions, on the **personborninlocation** dataset, a subset of NELL-995.

We found that after 100 iterations, the ReLU function beat out all the others, with an test-set MRR of 0.6191, closely followed by ReLU6 at 0.6161 and softplus at 0.6114. By the 300$^{\text{th}}$ iteration, ELU and cReLU, with MRR scores of 0.6217 and 0.6031 respectively, had overtaken ReLU which had a score of 0.5860.

It is clear that on this dataset, ELU was able to outperform ReLU when used in the hidden layers of the MLP policy network, after 300 iterations. The MRR of ELU at 300 iterations was greater than that of any other activation function in any iteration. This implies some degree of overfitting when training the network using ReLU activation, and all others except ELU. We believe that for this reason, care should be taken when choosing activation functions, and that the choice may need to be tuned for different datasets.

C. Disabling History Encoding

The LSTM network takes the history of entities and relations in a path, as well as other factors, as input, and feeds its output to the MLP policy network.

The authors themselves did an ablation study on the effectiveness of remembering path history, in which the agent chose the next action based on only local information (current state), without the history h_t. On the Kinship dataset, they reported a 27% decrease in Hits@1 performance and a 13% decrease in Hits@10 performance.

When we attempted the same ablation, our Hits@1 score decreased 76% and Hits@10 score decreased 41%. It is unknown why the difference in performance we observed was so much more drastic, even with In any case, it is clear that without the encoded history, MINERVA does a much poorer job of predicting answers, and this appears to be a key aspect of the model.

D. Simplifying MLP and LSTM Architecture

Das et. al cite an embedding dimension of 200 and hidden layer size of 400 for all of their experiments, although we encountered segfaults using their architecture. The authors did not justify the reason for their original architecture, so we explored simpler architectures to observe how the results differ. We compared MRR across different LSTM embedding dimension sizes and MLP hidden layer sizes for the Kinship and UMLS datasets.

With smaller architectures, MINERVA appeared to achieve nearly the same MRR scores as our original reproduced results. By reducing the hidden layer size, we were able to increase performance at earlier iterations, converging to the optimal parameters much faster than with a larger architecture. While it seems that a larger and more complex architecture may allow the model to ultimately perform better, smaller architectures are still viable and have a lower computational cost.

III. CONCLUSION

Our results indicate that the performance of MINERVA is not quite as strong as the authors of the original paper suggested, even after re-tuning hyperparameters. We have found that the authors' choice of ReLU as an activation function may not be optimal in all cases. However, it is clear that the design of the model cannot be significantly simplified, as ablating the history encoding or the neural network architecture both significantly decrease the effectiveness of MINERVA.

References

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.