Artificial Intelligence and Autonomous Systems 096208 - Abstract Submission

Gur Keinan 213635899, Naomi Derel 325324994

August 22, 2024

1 Field of Research

Our project aims to predict search progress utilizing Graph Neural Networks (GNNs). We define search progress as the fraction of the total search effort that has already been expended. More formally, given a search algorithm A and a heuristic search problem P, the search progress of algorithm A solving problem P after expanding $Gen_A(P)$ nodes with $Rem_A(P; Gen_A(P))$ remaining nodes to be expanded is defined as:

$$Prog_A(P) = \frac{Gen_A(P)}{Gen_A(P) + Rem_A(P; Gen_A(P))} \tag{1}$$

Meaning - the fraction of the total search effort that has already been expended.

We hypothesis that GNNs can learn the underlying structure of the search space and predict search progress more accurately than previous methods.

1. Consider a search node with a large immediate receptive field. We would expect it to have a lower search progress, as the search algorithm opens up many new nodes close to the current node, meaning it needs more exploration to reach the goal.
2. Consider a search node with a small immediate receptive field. We would expect it to have a higher search progress, as the search algorithm is likely to be close to the goal Gur: maybe rephrase this Omer: Naomi this is your color, you can call it using \naomi{text}

2 Literature Review

3 Work Domain

Our project will be practical, focusing on creating aduquate datasets and training GNNs to predict search progress. We aim to create our own simple dataset of Blocksworld Gur: citation problems, and use the A* search algorithm with different heuristics to solve them. We will then train a GNN to predict search progress based on the search nodes expanded by the A* algorithm.

Algorithm 1 Sampling Blocksworld Instance

Input: Number of blocks n

Output: Blocksworld instance

1: Initialize n blocks

2: Sample two permutations $p_1 = (i_1, i_2, \ldots, i_n)$ and $p_2 = (j_1, j_2, \ldots, j_n)$ of $1, 2, \ldots, n$.

3: Sample two binary vectors $b_1, b_2 \in \{0, 1\}^n$.

4: In the initial configuration, place block i_1 on the table. For every $k = 2, 3, \ldots, n$, place block i_k on block i_{k-1} if $b_1[k] = 1$, otherwise place it on the table.

5: In the goal configuration, place block j_1 on the table. For every $k = 2, 3, \ldots, n$, place block j_k on block j_{k-1} if $b_2[k] = 1$, otherwise place it on the table.

4 Work Outline

The following is a rough outline of our project:

1. Create a dataset of Blocksworld problems. An instance of the Blocksworld problem consists of a set of blocks, initial and goal configurations of the blocks. The goal is to move the blocks from the initial configuration to the goal configuration using legal moves. We hope that by sampling initial and goal configurations randomly uniformly, we will get a diverse set of problems that will be challenging for the search algorithm. We refer the reader to Algorithm 1 for a method to efficiently sample Blocksworld instances.

2. Implement the A* search algorithm with different heuristics to solve the Blocksworld problems. Without limiting ourself or guarantying to use them all, we plan to use the h_{max}, h_{add}, and h_n heuristics, where h_n is the number of blocks that are not in their correct position.

3. Running the A* algorithm on the Blocksworld problems, we will collect the labeled data for our GNN - for each node we expand, we will record the search progress.

4. Train a GNN to predict search progress based on the search nodes expanded by the A* algorithm. We will use the PyTorch Geometric library to implement the GNN.

5 Initial Ideas

Reasonable Results We expect that the GNN will be able to predict search progress more accurately than previous methods. Our first baseline wll be to make our results comparable (and as we hope, better) than the previous methods mentioned on Sudry and Karpas [2022]. Maybe we will implement the previous methods and compare them to our GNN, but this is not guaranteed.

Interpretability We hope that the GNN will be able to learn the underlying structure of the search space and provide insights into the search progress. We will analyze the GNN's predictions and try to understand what it has learned.

Generalization We hope that the GNN will be able to generalize to new Blocksworld problems that it has not seen before. We will evaluate the GNN on a test set of Blocksworld problems that it has not seen during training. We hope to also vary the number of blocks in the Blocksworld problems and see how the GNN generalizes to problems of different sizes.

References

Matan Sudry and Erez Karpas. Learning to estimate search progress using sequence of states. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 32, pages 362–370, 2022.