CS 703 Project Proposal

Xiating Ouyang

1 Motivation

Program synthesis aims at constructing a program which satisfies all the required specifications, and one common form of specification is the input-output examples. One trivial construction is to utilize the if-then-else statements to capture all input cases and return the corresponding output. However, this trivial approach is not scalable when all given examples conform to some function, since the number of cases captured by the synthesized program will be proportional to the size of the examples, while only one program implementing the function is sufficient. One synthesis approach is to synthesize all possible programs over every subset of the examples, select the minimum number of the synthesized programs that can cover all examples and finally combine the programs using the branching statements. Despite this approach potentially explores exponential many subsets, in practice however, it is efficient in certain circumstances since the input-output examples follow certain distributions. We hence wish to study different underlying distributions of the input-output examples based on which we explain the efficiency of program synthesis in certain circumstances.

Given a set of n input-output examples, we can construct a graph G as follows: introduce a vertex for each input-output example, and connect two vertices if there exists a program consistent with both examples. If there exists a program consistent on k examples, then their corresponding vertices form a clique of size k in G. Note that the edges in the graph may represent distinct programs. The largest number of examples that a program can be consistent on is hence no greater than the size of the maximum clique in the graph G. In addition, if the the maximum clique has $O(\log n)$ vertices, then the synthesis program only needs to explore all $O(n)$ subsets of these examples to search for consistent programs. Therefore, we focus on deriving appropriate probablistic graph models for the input-output examples. Then we will compute the expected maximum clique size to provide an explanation of the efficiency of our synthesis algorithm.

2 Preliminaries

All graphs discussed in this proposal are undirected and simple. A set of vertices in a graph is a *clique* if there is an edge between each pair of vertices within the set. A clique is *maximal* if it is not properly contained in another clique. A *maximum clique* is a clique with the maximum number of vertices. Note that a graph may have multiple maximum cliques. Given a graph G, the size of the maximum clique of G is called the *clique number*, denoted by $\omega(G)$.

3 Problem definition

The problem this project attempts to address can be formulated as:

> Let $\mathcal{G}$ be a probablistic graph model and a random graph G with $G \sim \mathcal{G}$. Compute $\mathbf{E}[\omega(G)]$.

Multiple random graph models have already been extensively studied. One classical example is the Erdős-Rényi model [3] in which each pair of vertices in a graph with n vertices has equal probability p of being adjacent. When $p = 1/2$, the clique number of G is at most $(2 + \varepsilon)\log_2(n) + 1$ with high probability, for all $\varepsilon > 0$ [4]. Power Law graphs characterize the networks in which the number of vertices with a certain degree k is propotional to $1/k^c$ for some constant c. This model is studied in [1], but an estimation of the size of the maximum clique is not present in the work.

4 Action plan

We propose to investigate the power law graph model and derive a bound for the size of the maximum cliques in the power law graphs for different parameter c. Our intuition is that if there are a set of

input-output examples consistent to one program, then examples consistent with any example in the set is likely to be consistent with the same program. If it is an appropriate model for the input-output example setting, our hypothesis is that the maximum clique size of a power graph model on n vertices with parameter c is $O(\log n)$ when $0 < c \leq c_0$ where c_0 is some constant, and that in reality c_0 should be relatively small. However, this still needs to be supported by proofs and experiments.

In addition to the power law graph model, we plan to find other random graph model for our synthesis setting either based on existing models or constructing a novel one. Hence we propose to research on the existing random graph models with interesting properties[2], and study the input-output examples from real data to build new models.

The goal of this project is to understand different random graph models and evaluate the applicability of each model to the input-output example setting.

5 Deliverables and milestones

The milestones for the projects are as follows with deliverables highlighted in **bold**.

- Oct 21: Finish implementing the **power law graph model**.
- Nov 1: Finish reviewing the literature on other models[1].
- Nov 12: Finish implementing and experiments on the **new model** if possible.
- Nov 18: **Proof sketch** of maximum clique size for power law graph model and new models if applicable.
- Dec 2: Finish **report** write-up and **slides** preparation.

References

[1] William Aiello, Fan Chung, and Linyuan Lu. A random graph model for massive graphs. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 171–180. ACM, 2000.

[2] David Easley and Jon Kleinberg. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2010.

[3] Paul Erdos and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5(1):17–60, 1960.

[4] Daniel A. Spielman. Random graphs: Markov's inequality, October 2007.

[1] A not so measurable milestone, but still critical.