**ABSTRACT**

**GATED GRAPH NEURAL NETWORKS VS RELATIONAL GRAPH CONVOLUTIONAL NETWORKS FOR THE VARIABLE MISUSE TASK**

**MUKUND RAGHAV SHARMA**

Identifying bugs in source code has been an extremely important part of software development since the inception of the industry. The majority of static analysis, the analysis of software without actually executing programs, is rule based without much involvement of deep learning until fairly recently. This paper engages in a comparative study of determining the more performant graph neural network model on the basis of test accuracy between Gated Graph Neural Network (GGNN) models and Relational Graph Convolutional Network (RGCN) models on the Variable Misuse Task, a prediction task involving choosing the correct variable based on all the variables of the same type in a particular scope.

The data is of source code from the files of 24 trending C# repositories that are converted into a modified Abstract Syntax Tree to represent a directed graph whose vertices that represent the tokens and relationships between the tokens are represented by edges. Each of these vertices are associated with one of the aforementioned type of networks for the training phase after a particular embedding is computed for each token.

The comparison to decide the more efficient model is based on the test accuracy of all the repositories, an esoteric repository and an extremely popular repository to cover the spectrum of different types of repositories. The results show that the RGCN based models outperformed the GGNN models for all cases, albeit, within < 5% range.

*Keywords: Deep Learning, Graph Neural Networks, Tensorflow, Sequence Models, Convolutional Models, Learning from Code, Static Analysis.*