# EE / CprE / SE 492 Weekly Report 05: SDMay21-29
# Intelligent Code Editor

Mar. 15 - Mar. 29
Client: Hung Phan
Advisor: Dr. Ali Jannesari

### Team Members & Roles:

Evan Christensen - Meeting Scribe
Ben Gonner - Report Manager
Jacob Puetz - Chief Engineer Software Systems
Jordan Silvers - Meeting Facilitator
Cory Smith - Test Engineer

---

### Weekly Summary:

In the previous week, work continued on line replacement within the plugin and a method of getting a classification from a line of text was created. Our client has been preparing more data for the team to convert into natural language, but we have not received the data at this point, so no progress has been made on that yet. Additionally, an Open-NMT network was trained on unprocessed data for our baseline, which created a model with approximately 42% accuracy. As for sorting the pseudocode, most classifications we want to have have been covered at this point, now we are working on sorting the data that slips through the defined categories.

### Past Week Accomplishments:

- GNN Predictions - Cory
    - A method has been developed to allow predictions to be made using the exported model of the trained GNN
        - The GNN we've been working with was designed to make all of its calculations using the adjacency matrix rather than the feature matrix, so making predictions by encoding words is impossible
        - Instead, our solution works by calculating the adjacency of the given piece of text and comparing its weights to every other document node to identify which one is the most similar, and return the prediction for that node instead
    - The solution has one main issue that makes it not ideal for use in our program
        - Calculating the mean-squared error of a node to every other node in a graph is extremely time consuming, and it can take nearly 15 seconds to get a prediction sometimes
        - I have optimized the process as much as I know how to, but this remains an issue to be solved
- Research on Line replacement - Ben/Jordan
    - Researched different ways to replace line text through visual studio event handler
    - Researched techniques that may have similar attributes to what we are doing

        - I.e. “find and replace” techniques could allow us to change lines but based on what we want
    - Research on more Stack Overflow forums for more techniques
- Sorting Training Data - Evan/Jacob
    - Most of the main categories we wish to classify are sorted out of the spoc dataset
    - Function declarations, loops, variables ect.
    - Now working on sorting the lines that make it through the defined classifications

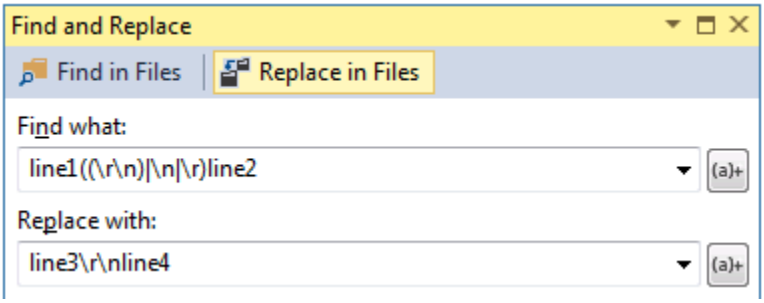
**Individual Contributions:**

| Name | Contributions | Hours This Week | Total Hours |
|---|---|---|---|
| Evan Christensen | Sorting Training Data | 3 | 20 |
| Ben Gonner | Line Replacement | 6 | 22 |
| Jacob Puetz | Sorting Training Data | 3 | 17 |
| Jordan Silvers | Line Replacement | 6 | 25 |
| Cory Smith | GNN Predictions | 6 | 44 |

**Upcoming Plans:**

- Experiment with an alternative system for getting a prediction from the GNN
    - I can train the GNN with an empty node in the adjacency matrix, and try to modify that when trying to make predictions
- Few other cases need to be added to the sorted data for training like cin and cout calls

**Examples:**




Similar in concept but without the find and replace box