10/21/2019 - 10/27/2019

E E/Cpr E/S E 491 Weekly Report 5

Intelligent Code Editor

Client & Advisor: Ali Jannesari
sdmay20-46
John Jago – Software Lead
Keaton Johnson – Systems Lead
Jon Novak – Machine Learning Lead
Matthew Orth – Meeting Facilitator
Garet Phelps – Report Manager
Isaac Spanier – Test Lead

Weekly summary

During this week the focus was on creating the java print dataset using the criteria we established last week. We also worked on context within the plugin, so it can determine if something is a variable, string, method, etc. OpenNMT-py was worked on, with focus shifting to the java print dataset.

Past week accomplishments

John Jago

- Plugin editor context
    - If a user says “print x”, for example, and if x refers to a variable, we want to print the value of x and not x as a string literal
    - Found a way to parse the contents of an open editor in Intellij and designed a small algorithm to perform the replacements
- n_best issue with OpenNMT-py server
    - Received a reply from maintainer of the project on GitHub. The feature is not currently supported but could be easily added. I will consider making a PR for this to their repository on GitHub.
- Architecture diagram
    - Created a diagram of the architecture of our current system
- Test plan (unit, integration, acceptance)
    - Began a document where we can describe in detail how we are testing the software

Keaton Johnson

- Dataset Generation
    - Wrote a program to generate random arithmetic dataset entries.
    - Generated 1500 lines of data

Jon Novak

- Worked on print dataset.
    - Made a program that would auto generate entries for dataset

Matthew Orth

- Java Print Dataset Creation:
    - Researched common Java print statement usages in the top GitHub Java projects
    - Added over 400 manual entries to the Java print training dataset taking into account the common GitHub usages and proper tokenization
- OpenNMT-py Implementation:
    - Experimented with effectiveness of different architectures, hyperparameters, and optimization on the OpenNMT-py system
        - Found larger batch size for RNN and Transformer architecture performs better
- Java Print Dataset Implementation:
    - Ran multiple (at various levels of complexity) Java print datasets through OpenNMT-py with the RNN and transformer architecture to determine baseline results to consider for dataset

Garet Phelps

- Worked on the Java print dataset
    - Made a program that generated dataset entries from a text file
    - Generated a large amount of entries.

Isaac Spanier

- Worked on the Java print dataset
    - Made sure to think about different ways to say print when generating my pseudocode to code translations.
    - Specifically tried to cover edge cases with my entries

Individual contributions

Name	**Contributions**	**Hours this week**	**Hours cumulative**

John Jago	● n_best issue with OpenNMT-py server ● Architecture diagram ● Test plan (unit, integration, acceptance) ● Plugin editor context	4	19
Keaton Johnson	● Dataset Generation ○ Wrote a program to generate random arithmetic dataset entries. ○ Generated 1500 lines of data	3	13
Jon Novak	● Worked on java print dataset ○ Made program to auto generate entries	5	14
Matthew Orth	● Java Print Dataset Creation ● OpemNMT-py Implementation ● Java Print Dataset Implementation	8	61
Garet Phelps	● Worked on the Java print dataset ○ Made a program that generated dataset entries from a text file ○ Generated a large amount of entries.	3	14
Isaac Spanier	● Worked on the Java print dataset ○ Edge Cases and Linguistics	3	12

Plans for the upcoming week

John Jago

- Continue working on extracting context from the current editor tab to make translations smarter about string literals versus variables references
- Update Engineering Standards and Practices section of the design documentation

Keaton Johnson

- Continue work on generating a dataset.
- Look into ways to automatically generate from GitLab repositories.

Jon Novak

- Continue working on print dataset and possibly move to more general datasets

Matthew Orth

- Run Java print dataset through the OpenNMT-py and record metrics

- Research Linguistics for Java print dataset

Garet Phelps:

- Work on the Java print dataset
- Research ways to automatically generate the dataset

Isaac Spanier

- Research Linguistics for the Java print dataset
- Look into GitHub's Search API to look for examples of the print usage

Summary of weekly client/advisor meeting

We met with Dr. Jannessari on 10/25/19. All members were present. We made a presentation to demonstrate what we had worked on. There was a lot of discussion on clarifying what tools we were using. He mentioned that we should diversify our natural language as much as possible in the dataset so that there is more flexibility in what can be converted to code. The model should also be self-learning. This next sprint will be improving and building on what we did during this sprint: expanding the database, adding context to the UI, and optimizing OpenNMT-py.