I. Conclusion

In this paper, we implemented a statistical-based code clone detector for Java files. As a counting based clone detector, it is simple and fast with a quadratic time cost. With the size being small, it maintains high precision and recall of over 90%.

To improve the detecting efficiency, we used machine learning to find the most proper weights. Machine learning can learn the developer's behavior, so users can use their own codes to train the tool. Its enhancement of efficiency is proved by actual testing.

Using SWT to develop a user interface makes STCD more user-friendly.

Overall in this paper, we built a MLP embeded framework to detect code clones in Java files, based on token frequency. It is efficient and aggressive in clone detection field.

Our future work includes the improvement of time cost by replacing ASTParser Tool with more simple methods, as well as enlarge the training and test data set to validate the result, and compare with other tools.