CHAPTER 1
INTRODUCTION

1.1 Background

Machine Learning has become the most rapidly growing technical field, becoming the core of Artificial Intelligence and data science [1]. Semi-Supervised Learning (SSL) is one of the Machine Learning Algorithms widely used to manage datasets where not all data have labels [2]. In SSL, a model is created using a small portion of the labeled data, and then the model is used to pseudo-label the unlabeled data. The pseudo-labels can then be added to the correct labels to update the model. However, one of the problems often encountered in SSL is a class imbalance, where there are more examples for specific classes in the dataset than others. This can cause the model to be overly affected by the majority classes, resulting in lower accuracy for minority classes or classes with fewer data.

Previous research has shown that models trained on biased data are more likely to favor the majority classes. Various techniques, such as re-sampling [3], two-stage training [4], and re-weighting [5], have been offered to decrease bias in supervised learning. However, while SSL has received much attention, it has received little attention regarding class-imbalanced data. A class rebalance method can be applied to overcome this problem, to balance the amount of data in each class and prevent the model from being overly influenced by the majority classes. One way to achieve class rebalance is to use a self-training framework that uses a basic SSL algorithm to pseudo-label the unlabeled data, then update the model by adding the pseudo-labels it deems most accurate to the correct set of labels. The Class Rebalancing Self-Training (CReST) is a self-training framework developed to address the problem of unbalanced learning in Semi-supervised learning. CReST works by pseudo-labeling the unlabeled data using the basic SSL algorithm, then updating the model by adding the pseudo-labels that it deems most accurate to the correct set of labels. However, CReST has an advantage over other self-training methods by updating the minority class more aggressively than the majority class during self-training iterations. It is intended that the model is not too influenced by the majority class and focuses more on the minority class. Yang and Xu [6] have claimed that using SSL and self-supervised learning to leverage unlabeled data can help with class-imbalanced learning. To further reduce the loss on minority classes, Hyun et

al [7] proposed a suppressed consistency loss technique. This technique aims to reduce the risk of a model being overly influenced by the majority class by suppressing the consistency loss for the majority class and allowing the minority class to impact the model significantly. The process that occurs in each generation of SSL after the implementation of CReST, what happens to the imbalanced data that allows it to be considered an algorithm for class rebalancing self-training, and the changes in pseudo-labeling in each generation are not yet thoroughly studied in previous research [8].

This undergraduate thesis aims to examine the processes in each generation of the class rebalancing self-training algorithm, which is applied to SSL with poor performance on imbalanced data. Specifically, the study aims to investigate the class rebalancing process and to prove that the minority class truly experiences aggressive sampling. The CIFAR-10 Long-tailed dataset, which has an imbalanced class distribution, will be used in this undergraduate thesis to determine the best generation. Using the class rebalancing self-training algorithm in Semi-supervised learning, which has problems with imbalanced class distributions, it is expected that CReST will aggressively update the minority class in its pseudo-labeling and that CReST will obtain the appropriate generation and good accuracy, precision, and recall.

1.2 Problem Formulation

Based on the background presented, the formulation of the problem in this Final Project are as follows.

1. There needs to be more evidence and further experimentation on the impact of different generations in the class rebalancing self-training framework for imbalanced semi-supervised learning.

2. Analysis of the effectiveness of the class rebalancing self-training framework for imbalanced semi-supervised learning towards classes with fewer data.

3. Analysis performance of the class rebalancing self-training framework for imbalanced semi-supervised learning by analyzing performance using precision, recall, and average recall accuracy rate for each class at each generation.

1.3 Objectives

Based on the results of the formulation of the problem, the objectives and benefits obtained in This Undergraduate thesis are as follows.

1. Understand the performance of each generation in class rebalancing self-training framework for imbalanced semi-supervised learning on the CIFAR-10 LongTailed dataset.

2. Performance Analysis for class rebalancing self-training framework for imbalanced semi-supervised learning on the CIFAR-10 LongTailed dataset.

3. Analyze and present data across generations based on Recall, Precision, and average accuracy-recall per class metrics.

1.4 Scope of Works

The scope of this undergraduate thesis is limited to the following:

1. The implementation of the study will be carried out solely in a software simulation environment using the CIFAR-10 Long-Tailed dataset.

2. The focus of the analysis in this undergraduate thesis is to observe the changes in each generation of the CReST Framework for imbalanced semi-supervised learning.

3. Author will apply two methods in implementing the CReST Framework for imbalanced semi-supervised learning: FixMatch and MixMatch. However, this study will only be using the FixMatch method.

4. The performance measurement of the CReST Framework for imbalanced semi-supervised learning will be limited to recall, precision, and the average accuracy recall for each class for each generation.

5. The number of generation parameters calculated is every five generations. The total number of stored generations is 30.

6. The programming language used for this study is Python with Tensorflow 2.10.0.

7. The operating system used for this study is Ubuntu Linux utilizing Google Colab.

8. The computer specifications used for system design and testing are a CPU of Intel Xeon, a GPU of Tesla T4 of 15GB, and a RAM of 13 GB.

1.5 Research Method

The research methods used in this thesis are as follows.

1. Literature Study

   At this stage, the author conducts research through journals, papers, online courses, and other theses related to SSL, imbalanced data, and the CReST framework for imbalanced SSL.

2. System planning

   In this stage, the author determines, downloads, and generates the required dataset with imbalanced classes, sets up the environment, and installs all requirements to run the CReST Framework for imbalanced semi-supervised learning.

3. Testing and analysis

   In this stage, the author conducts testing of the predetermined parameters and saves the output results of the training process for each generation. The author analyzes every five generations in this undergraduate thesis based on predetermined parameters.

4. Conclusion

   At this stage, the authors make conclusions based on the results of testing and analysis in the previous stage.

1.6 Undergraduate Thesis Organisation

The rest of this thesis is organized as follows:

- Chapter 2 BASIC CONCEPT
  This chapter consists of the Basic theory and related works of the CReST framework for imbalanced SSL.

- Chapter 3 SYSTEM DESIGN
  In this chapter, the system design is presented. It includes a flowchart that guides the author to conduct experiments on the CReST Framework for imbalanced semi-supervised learning. The process of selecting, downloading,

and generating the dataset, installing requirements and file dependencies, setting up the environment, and determining the parameters used in this undergraduate thesis is included.

- Chapter 4 PERFORMANCE EVALUATION
  This chapter discusses the precision, recall, and average recall accuracy results for each class at each generation. It also includes data visualization in graphs to aid in the analysis process.

- Chapter 5 CONCLUSIONS
  This chapter contains conclusions and suggestions for future development in this undergraduate thesis.