Research Instrument

System Architecture

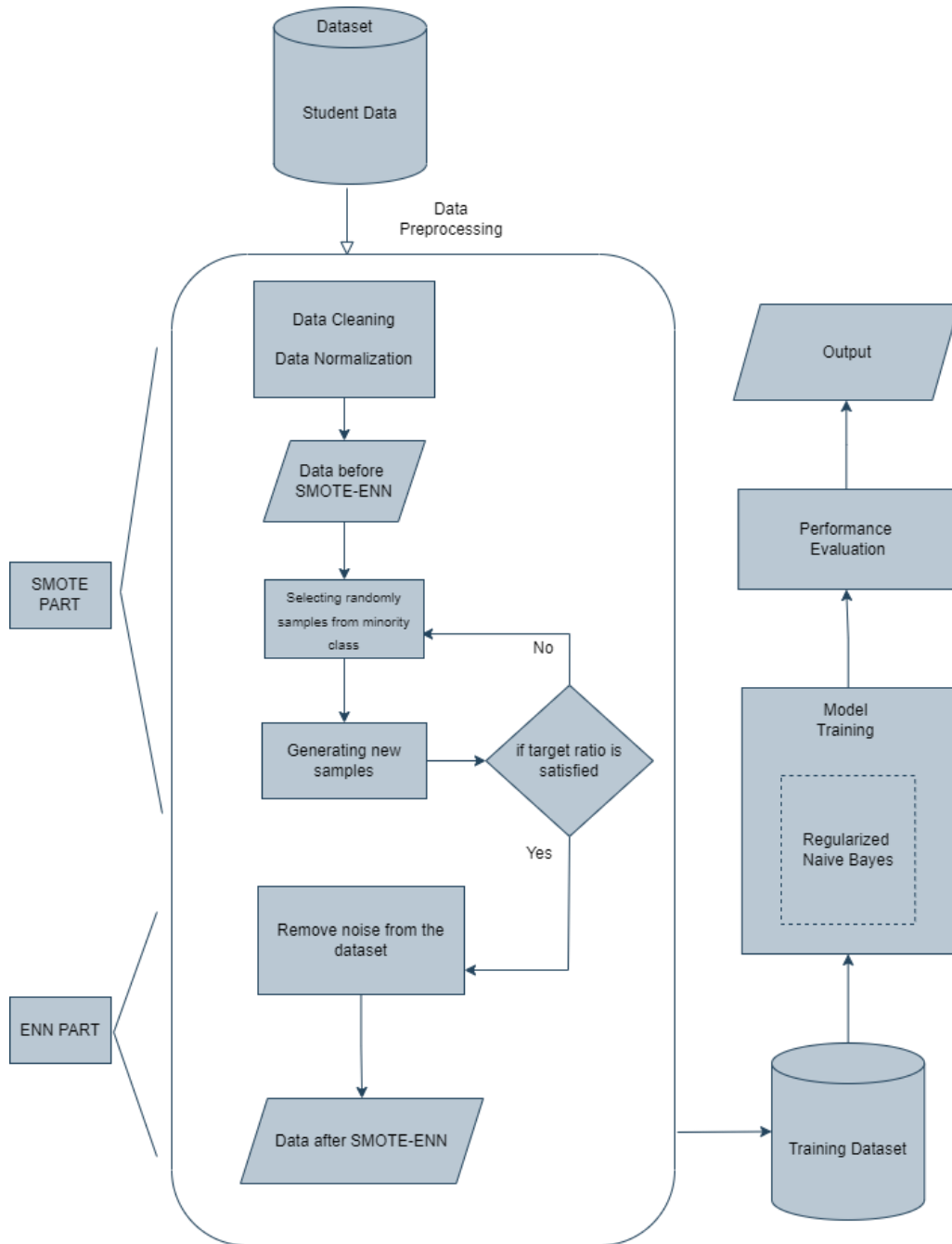

Figure #. System Architecture of our proposed model

Data Collection:

- Collect student records from the university database, including high school grades, senior high school grades, admission test results, and admission status.

Data Preprocessing:

- Clean the data to handle missing values and outliers.
- Normalize or standardize numerical data.
- Encode categorical variables into numerical format.

Handling Class Imbalance:

- Apply SMOTE (Synthetic Minority Over-sampling Technique) to generate synthetic samples for the minority class.
- Use ENN (Enhanced Neighborhood Cleaning Rule) to remove noisy and borderline instances.

Naive Bayes Model:

- Implement the Naive Bayes algorithm.
- Apply regularization techniques (L1 or L2) to prevent overfitting and improve generalization.

Evaluation:

- Evaluate the model using performance metrics such as accuracy, precision, recall, and F1-score.
- Use cross-validation to ensure robustness of the model evaluation.

Output: Admission Prediction:

- The final output is a prediction indicating whether a student is most likely to be admitted or not likely to be admitted.

Software Development Tools

This study will develop a system that will predict the admission of the applicants. The following tools will be utilized to create and improve the proposed system:

Hardware Tools:

- **CPU:** Intel i5-10400 (2.90GHz base clock)
- **Memory:** Kingston HyperX Fury 16GB (2666MHz)
- **GPU:** NVIDIA GeForce GTX 1660 Super 6GB Dedicated Memory

Software Tools:

- **Visual Studio Code (VS Code)** - is a well-known source code editor developed by Microsoft. It is an Integrated Development Environment (IDE) that runs on Windows, Linux, and MacOS. VS Code supports multitudes of extensions for other languages such as Python.
- **Git** – is a distributed version control system, allowing users to track versions of files. It is a powerful version controller for programmers that work collaboratively on developing software.
- **GitHub** – is an online platform that allows users to create repositories to store, manage, and share their code. It uses the Git software for managing file versions.
- **Python** – is a high-level programming language that is interpreted, object-oriented, and has dynamic semantics. Python supports modules and packages, enabling modularity and code reuse.
- **Python Libraries** – are packages/modules providing additional functionality, enabling the extension of tools or functions to source codes. The following libraries will be used for the accomplishment of the study:
    - **Scikit-Learn** – A widely-used Python library for machine learning that offers simple and efficient tools for data mining and data analysis, including implementations of various classification, regression, and clustering algorithms.
    - **Pandas** – A powerful Python library for data manipulation and analysis, offering data structures like DataFrames that make it easy to handle and analyze structured data.
    - **Imbalanced-learn** – A Python library that provides tools to address imbalanced datasets.
    - **NumPy** – A fundamental package for scientific computing with Python, providing support for arrays, matrices, and many mathematical functions to operate on these data structures.
    - **SciPy**