Final-Term Project
Submitted By:
Pendyala Hemanth Sai

Table of Contents

Introduction

In the field of data mining and machine learning, classification is a fundamental supervised learning task where the goal is to predict discrete labels for given input features. Binary classification, a specific type of classification, involves assigning one of two possible classes to each data instance. This project focuses on the application of binary classification techniques to a real-world medical dataset derived from molecular descriptors, which are commonly used in cheminformatics to represent the chemical structure and properties of compounds.

The dataset used in this study contains molecular descriptors generated using RDKit for chemical compounds labeled either as diabetic-related or non-diabetic-related. Each compound is represented by a set of numeric features describing its physicochemical properties and structural characteristics. The primary goal of this project is to predict whether a given compound is associated with diabetes using three different classification algorithms.

We implemented and compared three supervised machine learning algorithms: Random Forest, Support Vector Machine (SVM), and GRU(Gated Recurrent Unit). These algorithms were chosen for their varying strengths — Random Forest as a powerful ensemble-based model, SVM for its ability to find optimal decision boundaries in high-dimensional spaces, and GRU(Gated Recurrent Unit) for its interpretability and probabilistic outputs.

To evaluate the effectiveness of each model, we employed 10-fold cross-validation and calculated performance metrics manually based on the confusion matrix. Metrics include True Positive Rate (TPR), False Positive Rate (FPR), False Negative Rate (FNR), True Skill Statistic (TSS), and Heidke Skill Score (HSS). This comprehensive evaluation allows for a detailed comparison of classifier performance, with the ultimate objective of identifying the most suitable model for binary classification in this medical application.

Description of Project

This project involves binary classification using a real-world dataset derived from RDKit molecular descriptors. The classification task is to determine whether a compound is related to diabetes based on its chemical features.

The dataset includes two files: a training set and a test set. The training set is used to train three machine learning models, while the test set is kept aside for future use or external evaluation.

Three classification algorithms were selected for this project:

1. Random Forest

2. Support Vector Machine (SVM)

3. GRU(Gated Recurrent Unit)

All models are implemented using existing Python libraries, and the evaluation is done through manual metric calculation from the confusion matrix for each fold in 10-fold cross-validation.

Objectives

The primary objective of this project is to apply and compare the performance of different supervised classification algorithms for a binary classification problem using real-world medical data. Specifically, the goals are:

- To implement three classification algorithms: Random Forest, Support Vector Machine (SVM), and GRU(Gated Recurrent Unit) using existing Python libraries.
- To perform 10-fold cross-validation and evaluate the models based on manually calculated performance metrics.
- To assess model performance using evaluation metrics such as True Positives (TP), False Positives (FP), False Negatives (FN), True Negatives (TN), True Skill Statistic (TSS), and Heidke Skill Score (HSS).
- To identify the most accurate and reliable model for predicting diabetes-related compounds based on molecular descriptors.
- To analyze and interpret the results to understand which algorithm performs best and why.

Methodology

Dataset Description

The dataset used in this project consists of RDKit-generated molecular descriptors for chemical compounds. Each instance is represented by a wide set of numerical features capturing various structural and chemical properties. The target variable (`Label`) is binary, indicating whether a compound is related to diabetes (1) or not (0). The dataset was divided into a training set and a test set.

Data Preprocessing

- **Feature Selection**: The `SMILES` column was removed as it contains string representations of molecules and is not needed for modeling.
- **Label Conversion**: The target column was converted to integer format for binary classification.

- **Scaling**: Feature scaling was applied to the input data specifically for the Support Vector Machine (SVM), which is sensitive to feature magnitudes.

Model Implementation

Three models were implemented using Python and the `scikit-learn` library:

1. **Random Forest Classifier** – An ensemble method using multiple decision trees.
2. **Support Vector Machine (SVM)** – A classifier that seeks the optimal margin between classes in high-dimensional space.
3. **GRU(Gated Recurrent Unit)** – The Gated Recurrent Unit (GRU) is a deep learning architecture designed for sequence modeling tasks. It utilizes gating mechanisms to regulate the flow of information, effectively mitigating the vanishing gradient problem. GRUs offer a streamlined alternative to Long Short-Term Memory (LSTM) networks, often yielding comparable performance with reduced computational complexity.

Evaluation Strategy

Each model was evaluated using **10-fold cross-validation** to ensure robustness and reduce overfitting. For each fold:

- Predictions were made on the validation set.
- A **confusion matrix** was computed.
- From the confusion matrix, the following performance metrics were calculated manually:
    - True Positive Rate (TPR)
    - False Positive Rate (FPR)
    - False Negative Rate (FNR)
    - True Skill Statistic (TSS)
    - Heidke Skill Score (HSS)
    - Accuracy and Precision

Results were recorded in tabular format for each fold and averaged across all folds for comparison.

Implementation Summary

The project was implemented in a Google Colab notebook using Python. The implementation process included:

- Reading and preprocessing the dataset.

- Removing the SMILES (string-based) column and encoding the target labels.

- Applying feature scaling (only for SVM).

- Training each of the three classifiers using 10-fold cross-validation.

- For each fold, calculating confusion matrix components (TP, TN, FP, FN).

- Computing evaluation metrics manually: TPR, FPR, FNR, Precision, Accuracy, TSS, HSS.

- Tabulating results for each model and computing the overall averages.

- Visualizing the final metric comparison using bar plots.

All libraries used (e.g., scikit-learn, pandas, matplotlib) are publicly available

Running Code

Please open the attached link in google collab upload the dataset file into the environment and run the cells cell by
cell.
Source Code Link: Data Mining Final Term Project
Github Link:

Code Annotations

Code is modularized into function cells for ease of execution and clarity in Google Colab. Each function is responsible for a specific task:

`load_data()`: Loads and prepares the dataset.
`run_cross_validation(model, model_name)`: Executes 10-fold cross-validation for a given classifier.
`calculate_metrics(cm)`: Manually computes metrics from the confusion matrix.
`plot_metrics_table()`: Generates a tabular and graphical summary of model performance.

Every function is clearly annotated with inline comments to guide the reader through its logic. Results are presented both as printed tables and plots to aid interpretation.

Comparison between Algorithms

Which Algorithm to Choose:

Based on the average metrics across all 10 folds, the **Random Forest classifier** outperformed the other models in nearly every category. It achieved:

- The **highest accuracy and precision**, indicating fewer false positives and more reliable predictions.
- **Better recall (TPR)** compared to SVM and GRU(Gated Recurrent Unit), meaning it was more effective in correctly identifying positive (diabetes-related) compounds.
- **Superior HSS and TSS**, reflecting not only the ability to classify correctly but also accounting for class imbalance and random chance.

Why Random Forest Worked Best:

- **Handles high-dimensional data** effectively without overfitting.
- Is **robust to noisy or irrelevant features**, which are often present in molecular descriptor datasets.
- **Performs internal feature selection**, reducing reliance on extensive preprocessing.
- Can **capture non-linear relationships** in data, which might be critical in complex biological and chemical domains.

In contrast, SVM was sensitive to feature scaling and struggled in cases of class imbalance, often predicting only one class in some folds. GRU(Gated Recurrent Unit), although simple and interpretable, lacked the complexity to capture the underlying patterns in the data as effectively as Random Forest.

Conclusion

In this project, we evaluated and compared three supervised classification algorithms — Random Forest, Support Vector Machine (SVM), and GRU(Gated Recurrent Unit) — on a binary classification task involving medical data. Based on a thorough 10-fold cross-validation process and the calculation of detailed performance metrics, we observed differences in the classification capability of each algorithm.

Random Forest consistently performed the best in terms of overall accuracy and robustness, likely due to its ensemble nature and ability to handle high-dimensional data without much preprocessing. Support Vector Machine showed reasonable performance but was sensitive to feature scaling and suffered in cases with class imbalance. GRU(Gated Recurrent Unit), while easy to interpret, was generally less accurate compared to the other models.

These findings demonstrate the importance of model selection and evaluation methodology in data mining applications, especially in critical domains like healthcare. Future improvements may include hyperparameter tuning, advanced feature selection, and balancing techniques such as SMOTE to further enhance classification accuracy.