Thursday, April 3, 2025

CMPT 419 Project Report

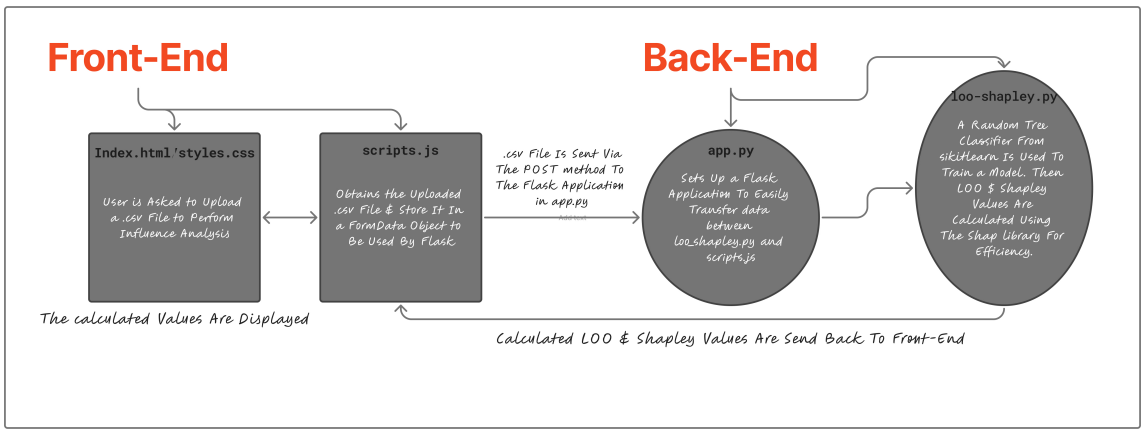
Arvin Bayat Manesh

- Abstract

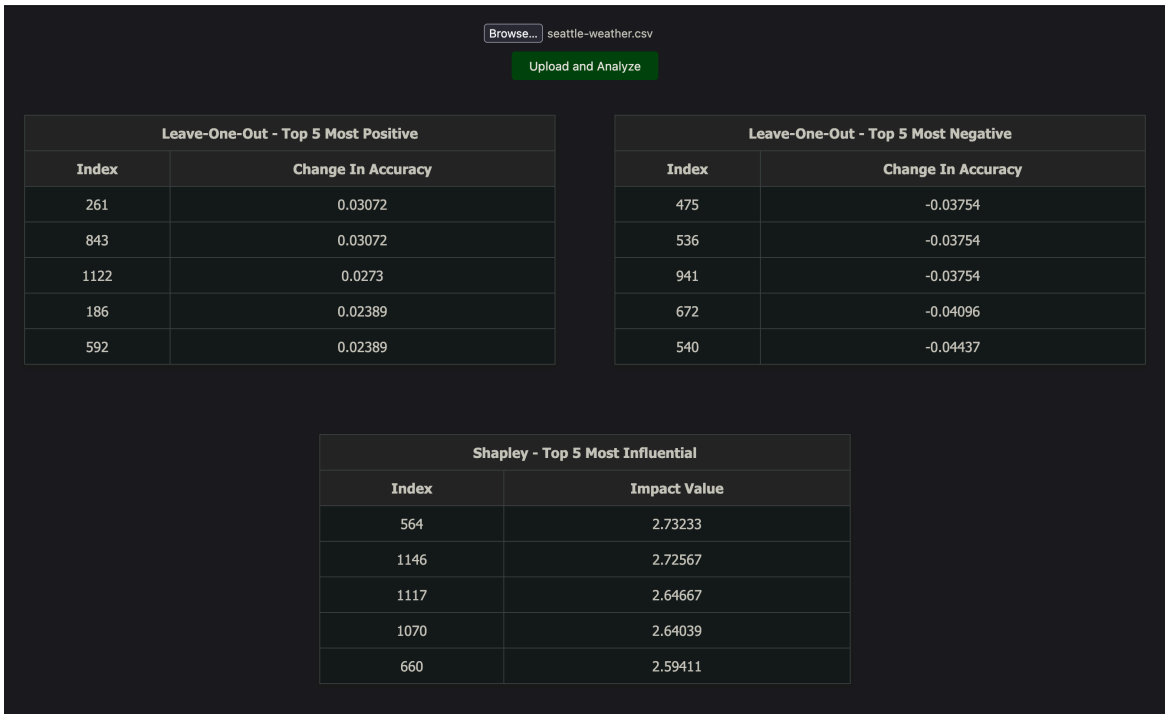
The machine learning pipeline can be divided into two main sections. The first is the development of the model, which uses various methodologies to learn the patterns existing in a dataset. The second is the methodologies and practices used to produce and curate the datasets that these models use to predict labels for unseen samples by identifying patterns in large-scale, high-dimensional data that are not easily apparent to the human eye.

While the AI community has historically focused heavily on model development, the creation of high-quality data remains essential for building fair, unbiased models that truly serve people. To encourage AI practitioners to closely evaluate the datasets used in model training, this project provides an interactive front-end tool that allows users to upload their datasets and identify key data points. Specifically, the tool returns the top 5 samples that most positively impact model accuracy, as well as the top 5 that most negatively affect it, by computing a Leave-One-Out (LOO) score for each instance. Additionally, it showcases the top 5 most influential samples based on their overall impact on the model's predictions by calculating Shapley values.

- Visual Artifacts



1 The Above Diagram Shows the Components of The Project & The Flow of Data



Leave-One-Out - Top 5 Most Positive	
Index	Change In Accuracy
261	0.03072
843	0.03072
1122	0.0273
186	0.02389
592	0.02389

Leave-One-Out - Top 5 Most Negative	
Index	Change In Accuracy
475	-0.03754
536	-0.03754
941	-0.03754
672	-0.04096
540	-0.04437

Shapley - Top 5 Most Influential	
Index	Impact Value
564	2.73233
1146	2.72567
1117	2.64667
1070	2.64039
660	2.59411

The Final Output of The Front-End When The User Uploads a Desired Dataset

- Introduction

Data production, acquisition, and curation remain vital in ensuring biased or poisonous data is not cascaded further down into the ML (Machine Learning) pipeline. It is also a crucial entry point into the ML loop to ensure the state of the world is not negatively affected by the predictions and decisions made by ML models. ML loop is an abstraction of how noisy, biased, and poisonous data points can negatively affect the state of the world if they are used to train ML models that make decisions and thus alter the state of the world. Then, data from the altered state of the world will be collected, and the biased datasets will be further cascaded into the ML pipeline.

This project offers a platform for AI and ML practitioners to interject in the data collection portion of the ML loop and enables them to look closer at data samples that negatively and positively affect the accuracy score of their models the most by calculating the Leave-One-Out

scores of all data samples. Additionally, this platform enables them to identify the most influential data samples that affected the model's prediction. These data could be informative or might be poisonous, or include biased information. Nevertheless, practitioners are able to use this platform to look closer at those specific data points and decide whether to keep them.

- Related Work

1. **Introduction of FairMl: https://fairmlbook.org/introduction.html**
   - Describes the ML loop and identifies the points of interjection to deter biased poisonous data from being cascaded down in the ML pipeline.
2. **Training Data Influence Analysis and Estimation: A Survey: https://arxiv.org/abs/2212.04612**
   - This article explains how to calculate Leave-One-Out (LOO) and Shapley values and how these values enable us to make informative decisions about whether to keep or discard a particular data point.
3. **SHAP Library to Efficiently Compute Shapley Values Using Mathematical Approximations: https://shap.readthedocs.io/en/latest/**
   - Calculating Shapley Values is an NP-complete problem and has a computation complexity of $O(2^n)$. The SHAP library uses approximation methods to efficiently compute Shapley values.
4. **Flask Back-End Framework To Transfer Data Between Front-End HTML to a Python File: https://flask.palletsprojects.com/en/stable/**
   - The Flask framework is used to implement efficient and straightforward communication between front-end scripts.js and back-end loo-shapley.py.

- Methods

This project aims to help users better understand the impact of individual data points on model performance by computing two influence metrics: Leave-One-Out (LOO) scores and Shapley values. This project offers an interactive web-based tool where users can upload labeled tabular datasets for analysis. The back-end system automatically trains a classification model on the data, computes per-instance influence scores, and presents the most impactful examples to the user.

Data Preprocessing

Before model training, standard data preprocessing tasks were performed. These tasks include one-hot-encoding of the labels and non-categorical features to ensure compatibility with the model being trained. Additionally, if the data set includes a feature that only includes unique values, such as a date feature where each instance corresponds to a specific day, that feature is removed as it does not contribute to the model's prediction ability.

Model Training

To ensure interpretability and compatibility with influence estimation techniques, a Random Forest Classifier from Scikit-learn was selected as the training model. Hyperparameters were set to scikit-learn’s defaults with the exception of the number of trees. The default number for trees used in the Random Forest Classifier is 100. However, it was reduced to 20 to make Shapley value calculation more efficient. Additionally, this decision was made because the focus of this project is to determine the most influential data points and not to train the most optimal model. Additionally, the model was chosen to be non-deterministic by not setting a fixed random seed. This approach introduces variability, aiding the user in identifying the data points that reoccur throughout different runs of the program. This idea helps to discard data points that have high Shapley or LOO values due to model variance or noise.

Leave-One-Out Value Computation

For computing LOO scores, the model is trained on the full dataset and then retrained for the number of data instances, each time excluding a single data point. The change in model accuracy is used to estimate the influence of that instance on model performance. While computationally intensive, this approach directly quantifies the contribution of each training point to the model's accuracy.

Shapley Value Computation

Shapley values were calculated using the SHAP library to assess the influence of individual data points on model predictions. For each training instance, SHAP computes feature attributions that quantify how much each feature contributed to a specific prediction. Specifically, the following lines of code:

```
explainer = shap.Explainer(base_model.predict_proba, X_train)
shapley_values = explainer(X_train)
shapley_values_array = shapley_values.values
```

produces a 3D NumPy array. These dimensions represent data instances, features, and classes or labels and include the calculated Shapley values for X_train. Each Shapley value represents how much a specific feature of a given data point influences the model's predicted probability for a particular class relative to a baseline probability. To obtain an overall influence score per data point, the absolute Shapley values across all features and prediction classes were aggregated. This aggregation yields a single value representing the total influence of each instance on the model's predictions. The implementation returns the five training examples with the highest aggregate influence scores.

Frameworks

The web interface was developed using the Flask framework. Users can upload datasets and receive interactive results showing the most positive and negative impactful samples and those with the greatest overall influence.

- Results

The main output of this project is a web-based platform for efficiently performing data influence analysis on a given dataset. After uploading a labeled tabular dataset through the web interface, users are presented with a ranked list of the most impactful training points according to both influence metrics described in the Methods section of the report. The tool supports CSV file uploads and automatically performs preprocessing, model training, and influence computation in the backend. The frontend displays results interactively, allowing users to examine:

1. The five most positively influential data points (These are the data points that increased the model's accuracy score the most)
2. The five most negatively influential data points (These are the data points that reduced the model's accuracy score the most)
3. The five data points with the highest influence on the model's prediction

- Discussion

Human-centered artificial Intelligence (HCAI) focuses on the interpretability of ML models and encourages transparency in ML practices. In this project, leave-one-out and Shapley values are used to determine influential data points in given datasets, aiming to highlight the importance of model explainability.

Specifically, the aggregated absolute Shapley values described in the methods section highlight how much a certain data instance and its features swerve the model's prediction probability away from or push the model's prediction closer to a baseline probability. In other words, imagine a particular model that predicts three classes, namely sunny, cloudy, or rainy. The base probabilities describe the probability of a particular class predicted on average. Let these probabilities be 0.4 for sunny, 0.35 for cloudy, and 0.25 for rainy. Now, let the features be A, B, and C. To calculate the Shapley value for feature A, we measure how much the probabilities differ from the base when we include {A}, {A, B}, {A, C}, and {A, B, C}. We then average the differences, calculate the Shapley value, and repeat for every instance feature and class. Therefore, the Shapley value aims to explain which data points are the most influential.

To support these interpretability goals, the Random Forest Classifier from scikit-learn was chosen as the base model. However, it should be noted that the choice of Random Forest does not limit the generality of the influence metrics used. Shapley values are model-agnostic, which means that the underlying calculated values are not unique to the specific model chosen but instead reflect the relationship between data features and predicted outcomes.q

Furthermore, the decision not to fix a random seed to produce non-deterministic models was made to determine training examples that appear influential across multiple model instantiations. Consistently high influence across different runs provides a stronger indication of the data point's impact, while appearing only once may suggest influence due to variance or noise.

In addition to HCAI, this project also aims to address Data-Centric Artificial Intelligence (DCAI) by identifying data points that could be mislabeled, poisonous, or included by an adversary. For instance, high aggregated Shapley values in the context of this project do not indicate whether the high impact is positive or negative. Data practitioners are encouraged to use this platform to look closer at these data points that could potentially harm the model's performance. Additionally, the data points that reduce the model's accuracy score the most could be outliers or mislabeled data. This platform enables the users to identify these points much quicker.

References

Barocas, S., Hardt, M., & Narayanan, A. (2023). *Fairness and machine learning: Limitations and opportunities*. https://fairmlbook.org/introduction.html

Hammoudeh, Z., & Lowd, D. (2022). Training data influence analysis and estimation: A survey. *arXiv preprint arXiv:2212.04612*. https://arxiv.org/abs/2212.04612

Github Repository Link: https://github.com/arvinbm/DATA_INFLUENCE_ANALYSIS