Applied Analytics Project - Week 10 Assignment

Data Centric AI

Brady Levenson
ADAN 8888 - Applied Analytics Project
Professor Nurtekin Savas
November 10, 2024

Revisitation of Problem Statement for Motivation to Perform Data Centric AI

It is known that fraudulent financial transactions occur on a regular basis, but it is difficult to distinguish between them and legitimate financial transactions simply by viewing the transactions. The inability to correctly classify fraudulent financial transactions can promote uncontrolled and untraceable levels of financial losses for banks, investment firms, corporations, and numerous other entities. We are given a dataset of financial transactions and their unique properties. We will analyze this dataset using various machine learning techniques to determine effective measures to identify historical fraudulent transactions and prevent their occurrence in the future.

Purpose of Performing Data Centric AI

Data-centric AI can involve data management strategies inclusive of ensuring the data is reflective of the real world problem it is being leveraged to solve, weighing data from multiple sources to add diversity, and augmenting synthetic data when appropriate, among other data improvement techniques. By reengineering a dataset, it is possible that an already existing model may be able to perform better [or worse] than it could have before, allowing a data scientist to identify data-related factors that are most contributory to a model's predictive capabilities. This may present insights leading to further model-centric AI developments that can result in optimized dataset-model combinations.

Three New Ways Used to Improve the Data

SMOTE - Synthetic Minority Over-Sampling Technique

SMOTE generates synthetic data points possessing similar attributes to the minority class of an imbalanced classification problem. The purpose of SMOTE is to balance the classes in this classification task so the minority class can be analyzed more substantively than it could have before. The application of SMOTE in the context of this unsupervised learning financial anomaly classification project took some additional consideration to be properly implemented. We first needed to label the dataset. Using our winning K-Means clustering model from Week 9, we identified clusters that were below the 10th percentile of size from this model (excluding clusters of size 1 that may be indicative of "noise"). We then placed a unique label on these transactions as compared with the rest of the transactions so that SMOTE could be applied to the smaller cluster data points that may justifiably be placed under anomaly consideration. Creating new samples

mirroring elements of our smaller clusters of samples may support a more balanced consideration of these smaller clusters of samples when cross-analyzed with the samples falling in clusters in the top 90% of cluster size identified by the model.

PCA - Principal Component Analysis

Principal Component Analysis is applied in this context to identify linear combinations between features in our given datasets while retaining 95% of the variance present in this data. In the original data cleaning and feature engineering sections of this project I had considered performing PCA on our data but decided against it as I had already performed significant dimensionality reduction operations on the data and figured further dimensionality reduction could possibly be detrimental to model performance. Now that we are at a stage of optimizing the datasets further to improve model performance, I see much more reason to implement this step now.

Gaussian Noise Injection

Gaussian Noise Injection is a regularization technique that artificially adds noise to the dataset at hand with the intent of forcing a given model to generalize to make conclusions rather than memorizing patterns in the data. This technique makes the dataset less predictable for more or less sophisticated models and better stimulates real world data where the presence of noise is almost always unavoidable. This data centric AI technique may also divert attention from noise that is already present in our datasets, which may allow our winning model to achieve even more robust insights based mainly on relevant data.

Error Analysis and Types of Data Improvements as a Result

Week 9's winning model achieved a silhouette score (primary evaluation metric for this task) of 0.4865 on a scale from -1 to 1, which indicates that the data is moderately well-clustered and has room for improvement in quality of clustering. **SMOTE** was applied to our train dataset to gain a better representation of minority class data. The intent of this was to improve clustering by introducing newly clustered data based on already prospective anomaly clusters, potentially addressing the issue of "normal" clusters overlapping with "anomaly" clusters. **PCA** was conducted to further reduce the dimensionality of our data in hopes of better capturing overall trends and underlying patterns in our data, which may result in the production of more distinct and relevant clusters. **Gaussian Noise Injection** was executed to stimulate uncertainty in the data to promote

the creation of diverse and distinct clusters. By introducing Gaussian Noise, it may be possible for a model to gain a clearer perspective on where distinct cluster boundaries should lie, resulting in more precise cluster structures.

Refitting Week 9 Winning Model on New Train Dataset

After performing data centric AI techniques on the train dataset, we need to refit the Week 9 Winning Model on our new train set so the model can uncover patterns specifically from this dataset and apply these findings to our new validation and test datasets.

```
# Fit the k-means model again on the final data
kmeans = KMeans(n_clusters=10, init='k-means++', n_init=1, max_iter=200, tol=1e-3, random_state=42)
kmeans.fit(X_train_final)  # Fit on the final transformed train data
```

Week 10 Model Performance Vs. Week 9 Model Performance

Week 10:

Winning Model Performance Metrics by Stage on Data-Centric AI-Transformed Data

	Model Stage	Silhouette Score
0	Train	0.2975
1	Validation	0.31
2	Test	0.3083

Week 9:

Winning Model Performance Metrics by Stage

	Model Stage	Silhouette Score
0	Train	0.4638
1	Validation	0.4909
2	Test	0.4865

Based on model performance on the data centric AI-transformed validation dataset vs the original validation dataset, our winning model's performance **decreased** from Week 9 to Week 10 (silhouette score of 0.4909 in Week 9 compared to 0.3100 in Week 10). The lower silhouette score indicates poorer clustering of the transaction data, making it more difficult to identify anomaly transactions from this model. I believe the Week 9 Model outperformed the Week 10 Model because the data centric AI techniques applied to our data may have introduced irrelevant data and

performed too much dimensionality reduction. The intent of performing SMOTE on our data was to further analyze anomaly transactions by synthetically creating more of them, though selecting the smallest 10% of clusters. Since this model's hyperparameter tuning calls for the creation of 10 distinct clusters, the smallest 10% of these clusters represents the smallest of the 10 clusters. Since it is very likely not the case that all of the data points in the smallest cluster are truly fraudulent financial transactions, SMOTE is essentially introducing a combination of normal and anomaly data points and mislabeling some normal points as anomalies. Beyond SMOTE, it is possible that PCA may have reduced the dimensionality of the dataset in a detrimental manner. We already had a dataset with extensive one-hot encoding, embedding, and feature engineering, so it is possible that performing PCA on top of this may have taken away some of the interpretability of this data.

Final Model to be Deployed: Week 9 Winning Model

Winning Model Performance Metrics by Stage

	Model Stage	Silhouette Score
0	Train	0.4638
1	Validation	0.4909
2	Test	0.4865

Once again, here are the Train, Validation, and Test Performance Metrics of our winning model from Week 9. When the model was fit on the train dataset, it achieved a silhouette score of 0.4638. When applied to the validation dataset, the model did a good job of generalizing and achieved an even better score of 0.4909. When faced with the unseen test dataset, the model achieved a comparable score of 0.4865 to its performance on the validation dataset, which may be attributable to slight variations in the data rather than in the model's ability to make predictions on new, unseen data.

Insights from Test, Validation, and Training Errors

Our winning model's errors on test, validation, and training datasets are indicative of this model clustering financial transaction data moderately well. At this point, I would consider this model to be serviceable in some capacity at successfully identifying anomaly financial transactions. The decrease in model performance experienced when transitioning from the Week 9 Data to the Week 10 Data can be used to gain additional insights on what factors the model grasps best from a given dataset, which can be used as motivation for performing different combinations of data centric AI

techniques to ultimately attain even better performing models. Overall, our winning model does an effective job at isolating groups of financial transactions that should be isolated and evaluated further to determine their true likelihood of being fraudulent financial transactions. Further combinations of data centric AI and model centric AI techniques may result in us having more confidence in this model having purpose as a real world fraud detection agent.

Next Steps

Our next step in the end to end model development process will be explaining our selected model in depth and analyzing the relevant risk, bias, and ethical considerations surrounding its deployment in the real world. This will yield us with better perspectives as to whether or not this model should be purposed for real world deployment or if it is best to be kept in theoretical use.