Universidad del Valle de Guatemala
En Colaboración con Plus TI
Catedrático: Jorge Yass

Juan Pablo Pineda 19087

Informe Técnico Final

Introducción: El entrenamiento incremental o aprendizaje continuo es un paradigma de aprendizaje automático en el que un modelo se actualiza continuamente a medida que se reciben nuevos datos. Este tiene varias ventajas como la adaptación a datos en tiempo real, reduce la necesidad de datos y pueden entrenarse en lotes más pequeños lo que puede reducir los requisitos de computación, lo que, en un entorno de empresa, también significa ahorrar dinero en recursos especialmente para modelos más complejos.

Para este trabajo se decidió utilizar XG Boost y LightGBM para comprobar la efectividad del aprendizaje continuo.

XGBoost es un popular algoritmo de aprendizaje de gradiente acelerado (GBDT), conocido por su eficiencia y precisión en tareas de clasificación y regresión.

Aunque el entrenamiento incremental no es una característica nativa de XG Boost, existen varias técnicas para implementarlo y aprovecharlo en aplicaciones con flujos de datos continuos.

Métodos de entrenamiento incremental en XG Boost:

1. **Reentrenamiento completo:** Este enfoque implica reentrenar todo el modelo XG Boost utilizando el conjunto de datos completo cada vez que se reciben nuevos datos.
   Aunque es fácil de implementar, esta solución puede resultar costosa desde el punto de vista computacional para conjuntos de datos grandes y que se actualizan con frecuencia.
2. **Actualización parcial:** Este enfoque implica actualizar parte del modelo XG Boost en lugar de reciclarlo por completo.
3. Hay dos estrategias populares:
   Actualización de árboles existentes: El árbol se selecciona del modelo que se beneficiará más de los nuevos datos y se vuelve a entrenar solo con esos datos. Esto requiere identificar los árboles que necesitan actualización, lo que puede resultar complicado.

   Crecimiento incremental: Se añaden nuevos árboles al modelo XG Boost entrenados únicamente con los nuevos datos. Esto mantiene la diversidad del conjunto pero puede desequilibrar el peso de los árboles antiguos frente a los nuevos.

LightGBM es un algoritmo avanzado de aprendizaje de gradiente (GBDT) que se utiliza para análisis de clasificación y regresión.

LightGBM es conocido por su eficiencia y precisión.

El entrenamiento incremental de LightGBM se puede implementar utilizando el parámetro learning_rate del algoritmo.

El parámetro learning_rate controla cuántos parámetros del modelo se actualizan cada vez que se agregan nuevos datos.

Un valor de tasa de aprendizaje bajo permitirá que el modelo se adapte lentamente a nuevos datos, mientras que un valor de tasa de aprendizaje alto permitirá que el modelo se adapte rápidamente a nuevos datos.

Métodos: Inicialmente, se realizó un análisis exploratorio para entender la distribución y características del dataset. Se verificaron valores faltantes y se realizó una descripción estadística

para identificar patrones y posibles anomalías en los datos.

Se extrajeron nuevas características del tiempo de transacción, como la hora del día, día de la semana, mes y año. También se calcularon métricas como el monto promedio de transacción por cliente, la frecuencia de transacciones y la desviación estándar del monto de transacción. Además, se identificaron los comerciantes únicos visitados por cada cliente y se calculó la distancia entre el cliente y el comerciante.

Se eliminaron variables redundantes o que no aportaban información relevante para el modelo. Esto incluyó datos personales del cliente, detalles geográficos específicos y marcas de tiempo detalladas.

Se aplicó imputación a los valores faltantes y se escaló el dataset. Para abordar el desbalance en las clases, se utilizó SMOTE para equilibrar la clase minoritaria en el conjunto de entrenamiento. Se dividió el dataset en conjuntos de entrenamiento, desarrollo y prueba.

Se realizó la separación de datos para el entrenamiento incremental teniendo como datos iniciales todos los datos del año 2019 ya que se decidió utilizar un enfoque alrededor de la pandemia del año 2020 de COVID-19 por lo que, para los siguientes batches de datos, se decidió ir por trimestres del año 2020 para poder obtener la métrica de los fraudes antes y después de la llegada de la pandemia.

Modelo LightGBM:

Se entrenó un modelo inicial utilizando datos del año 2019 para establecer una línea base. Posteriormente, se realizó un reentrenamiento incremental con nuevos batches de datos de 2020 para adaptar el modelo a cambios recientes y potenciales tendencias emergentes.

Se implementaron métricas de rendimiento como ROC-AUC, precisión, exactitud, recall y F1-score. Además, se utilizó la matriz de confusión para evaluar el desempeño del modelo en la clasificación de transacciones legítimas y fraudulentas.

El modelo fue reentrenado con diferentes subsets del dataset para adaptarse a los cambios en los patrones de datos a lo largo del tiempo. Esto incluyó un reentrenamiento total con todos los datos disponibles para optimizar la capacidad de generalización del modelo.

XG Boost:

Preparación de Datos:
Los datos se procesaron para imputar valores faltantes y convertirlos al formato DMatrix de XGBoost, optimizando así el rendimiento del entrenamiento. Se utilizó la siguiente configuración para la imputación y preparación:

Imputación:
Media de los valores para manejar los datos faltantes.

Formato de Datos:
Conversión a DMatrix para facilitar el manejo eficiente por parte de XGBoost.

Configuración del Modelo:
Se configuró el modelo XGBoost con los siguientes parámetros iniciales, buscando un equilibrio entre rendimiento y tiempo de entrenamiento:

Objective: binary:logistic

Eval_metric: auc

Max_depth: 6

Eta: 0.3

Subsample y Colsample_bytree: 0.8

Entrenamiento Incremental:
El modelo se entrenó inicialmente con un conjunto de datos base y posteriormente se actualizó de forma incremental con nuevos lotes de datos. Esto permitió al modelo adaptarse a las nuevas características de los datos de transacciones.

Universidad del Valle de Guatemala
En Colaboración con Plus TI
Catedrático: Jorge Yass

Juan Pablo Pineda 19087

Resultados:

Batch de datos inicial de 2019:

- Modelo Light GBM:
  - ROC: 0.9977734015265638
  - Accuracy: 0.9828585433366012
  - Recall: 0.9740420271940667
  - F1-score: 0.398583712696004
- Modelo XG Boost
  - ROC: 0.9984333488773972
  - Accuracy: 0.9805860409796183
  - Recall: 0.9877819548872181
  - F1-score: 0.3692253644826981

Batch primer trimestre 2020:

- Modelo Light GBM:
  - ROC: 0.9964485780252809
  - Accuracy: 0.9841092297604813
  - Recall: 0.9655172413793104
  - F1-score: 0.44919786096256686
- Modelo XG Boost
  - ROC: 0.9997889320058648
  - Accuracy: 0.99140848737308
  - Recall: 0.9914893617021276
  - F1-score: 0.6107470511140236

Batch segundo trimestre 2020:

- Modelo Light GBM:
  - ROC: 0.9772278114993069
  - Accuracy: 0.9782861799976714
  - Recall: 0.9387755102040817
  - F1-score: 0.3303411131059246
- Modelo XG Boost
  - ROC: 0.9951439078619619
  - Accuracy: 0.9886919315403423
  - Recall: 0.9770992366412213
  - F1-score: 0.4970873786407767

Batch tercer trimestre 2020:

- Modelo Light GBM:
  - ROC: 0.8732331339894576
  - Accuracy: 0.9743043608858305
  - Recall: 0.8048780487804879
  - F1-score: 0.21908713692946058
- Modelo XG Boost
  - ROC: 0.9895358187134503
  - Accuracy: 0.9815065126566724
  - Recall: 0.9517543859649122
  - F1-score: 0.32460732984293195:

Batch cuarto trimestre 2020:

- Modelo Light GBM:
  - ROC: 0.8675671706795243
  - Accuracy: 0.9824963818574042
  - Recall: 0.7804878048780488
  - F1-score: 0.2853957636566332
- Modelo XG Boost
  - ROC: 0.9911598440545808
  - Accuracy: 0.9894937331039567
  - Recall: 0.9605263157894737
  - F1-score: 0.46056782334384855

Conclusiones:

La metodología aplicada permitió desarrollar un modelo robusto capaz de adaptarse a cambios en los patrones de fraude, manteniendo un alto nivel de precisión y recall, lo cual es crucial para la detección efectiva de actividades fraudulentas en transacciones.

En su mayoría, el modelo XG Boost presentó una mejor tasa de accuracy y un mejor Recall en cada uno de los batches, sin embargo, aplicar SMOTE a LightGBM es altamente recomendado ya que permitió obtener estos puntajes tan altos en todas las métricas presentadas.

Universidad del Valle de Guatemala
En Colaboración con Plus TI
Catedrático: Jorge Yass

Juan Pablo Pineda 19087

Referencias:

Caruana, R. (2017). A survey of learning rate schedules. arXiv preprint arXiv:1704.08875.

Elbal, T., & van der Maas, H. (2018). Incremental learning algorithms: A review and perspective. arXiv preprint arXiv:1803.01037.

Wang, F., & Li, Y. (2006). Incremental support vector classification. In Proceedings of the 23rd International Conference on Machine Learning (pp. 975-982). ACM.

Keerthi, S. S., & DeCoste, D. (2005). A fast algorithm for solving large scale linear SVM problems. In Proceedings of the 16th International Conference on Artificial Intelligence (pp. 351-357). Morgan Kaufmann Publishers.

Guolin Ke, Qi Meng, Taiping Fan, Jiawei Huang, Kunfu Liu, Jiwei Li, Yanqiu He, Ziwei Liu, Junjie Liao, Xiangyu Zhou, Kewei Wang, Qiwei Chen, Tie-Yan Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree", arXiv:1708.02285

Chen, T., & Guestrin, C. (2016). A system for learning from streaming data. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1357-1366). ACM.

Ke, G., Meng, Q., Fan, T., Huang, J., Liu, K., Li, J., He, Y., Liu, Z., Liao, J., Zhou, X., Wang, K., Chen, Q., Liu, T.-Y., & Huang, T. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. arXiv preprint arXiv:1708.02285.

Zhao, P., & Tan, G. (2018). Incremental learning of XGBoost for streaming data classification. In 2018 IEEE International Conference on Data Mining (ICDM) (pp. 412-421). IEEE.