Plus Ti

Convenio PLUS TI – Universidad del Valle

Trabajo Práctico Grupo 2: “Entrenamiento Incremental en Modelos de Deep Learning y Machine Learning”

Objetivo

Este trabajo práctico tiene como objetivo investigar la viabilidad del entrenamiento incremental en modelos de aprendizaje automático y profundo, utilizando como estudio de caso un dataset de transacciones de tarjeta de crédito clasificadas en normales y fraudulentas. Los modelos a investigar den incluir 2 de los siguientes algoritmos: Redes Neuronales Artificiales (ANN), LightGBM, XGBoost, Random Forest y Máquinas de Vectores de Soporte (SVM).

Adicionalmente, se debe establecer qué metodologías son las recomendables para determinar cuándo es preferible un reentrenamiento total frente a uno incremental.

Descripción del Dataset

Este es un dataset simulado de transacciones con tarjeta de crédito que contiene transacciones legítimas y fraudulentas desde el 1 de enero de 2019 hasta el 31 de diciembre de 2020. Cubre tarjetas de crédito de 1000 clientes que realizan transacciones con un conjunto de 800 comercios. Cuenta con 23 variables originales.

Parte 1: Entrenamiento Incremental

1. Investigación Teórica: Realizar una revisión bibliográfica sobre el entrenamiento incremental en los algoritmos seleccionados, destacando las capacidades y limitaciones existentes.

2. Implementación Práctica: Utilizando sus notebooks o Google Colab y bibliotecas de Python de su elección, implementar versiones incrementales de ANN, LightGBM, XGBoost, Random Forest y SVM. Deben entrenar inicialmente los modelos con un subset del dataset y luego aplicar entrenamientos incrementales con batches más pequeños y recientes. Se debe documentar cualquier modificación necesaria para habilitar el entrenamiento incremental en cada modelo. Para esto deberán partir el dataset en Train, Dev y Test.

3. Evaluación: Comparar el rendimiento de los modelos en términos de ROC-AUC, precisión, recall y F1-score, antes y después del entrenamiento incremental. Analizar si hay pérdida significativa en la capacidad de detección de transacciones fraudulentas.

Parte 2: Criterios para Reentrenamiento

1. Desarrollo de Metodología: Basándose en la literatura y en los resultados experimentales, proponer una metodología para decidir cuándo realizar un reentrenamiento total en lugar de uno incremental. Considerar factores como la variación en el rendimiento del modelo, el tiempo desde el último entrenamiento total, y la aparición de nuevas tendencias en los datos.

2. Validación Empírica: Aplicar la metodología propuesta al conjunto de modelos entrenados, justificando decisiones de reentrenamiento total o incremental con base en los criterios establecidos.

Aspectos Adicionales para el Trabajo Práctico

Análisis Exploratorio de Datos (EDA): Antes del entrenamiento de modelos, realicen un análisis exploratorio para entender las características del dataset, incluyendo visualizaciones, análisis de correlaciones y la distribución de las clases.

Feature Engineering: El feature engineering es crucial. Deben explorar la creación de nuevas características que puedan incluir, pero no se limiten a, variables temporales (como la hora del día, día de la semana, estacionalidad), frecuencia de transacciones por cliente, montos promedio de transacción, y diversidad de comercios visitados por cada tarjetahabiente.

La normalización o estandarización de estas características también es importante dependiendo del algoritmo. Para esta tarea de feature emgineering explicaremos como realizarlo y además entregaremos un script modelo en Python.

Manejo de Datos Desequilibrados: Explore técnicas como oversampling, undersampling, o generación de datos sintéticos (ej., SMOTE) para manejar el desequilibrio en la clasificación de transacciones normales y fraudulentas.

Afinación de Hiperparámetros: Experimente con la optimización de hiperparámetros para mejorar el rendimiento de los modelos, tanto en entrenamientos iniciales como incrementales.

Estrategias de Early Stopping: Implemente early stopping en modelos de deep learning para prevenir el overfitting y reducir el tiempo de entrenamiento, crucial en escenarios de entrenamiento incremental.

Métricas Específicas para Datos Desequilibrados: Utilice métricas como ROC-AUC o la curva de precisión-recall para evaluar de manera más efectiva el rendimiento del modelo en el contexto de datos desequilibrados.

Entrega

El trabajo deberá entregarse en forma de un informe técnico de no más de cuatro páginas, acompañado de los notebooks utilizados para la implementación y los resultados de los experimentos. El informe debe incluir:

- Resumen de la investigación teórica sobre entrenamiento incremental.
- Descripción de la implementación práctica, incluyendo ajustes realizados para habilitar el entrenamiento incremental.
- Análisis de los resultados de la evaluación, con énfasis en la detección de transacciones fraudulentas.
- Presentación de la metodología propuesta para decidir entre reentrenamiento total o incremental, con ejemplos de aplicación práctica.
- Conclusiones y recomendaciones para futuras investigaciones o aplicaciones prácticas.

Plus Ti

Evaluación

El trabajo será evaluado en base a la profundidad de la investigación teórica, la calidad técnica de la implementación, la rigurosidad del análisis de resultados, la viabilidad y originalidad de la metodología de reentrenamiento propuesta, y la claridad de la comunicación escrita.

Recursos Adicionales

Se puede optar por utilizar Google Colab para facilitar el uso de recursos computacionales, incluyendo GPU si es necesario. Se recomienda el uso de bibliotecas de Python como Scikit-learn, TensorFlow, o PyTorch para la implementación de los modelos.

Link descarga Dataset y Script Feature Engineering:
https://drive.google.com/file/d/19L8VAElj8kguuwWDwbSgsjeOUWjUFvUk/view?usp=sharing

info@plusti.com 706 257 - 6555

1921 Whittlesey Road Suite 500 Columbus, Georgia 31904