Machine Learning para preservar la anonimidad

Normalmente las técnicas de machine learnign deben manegar volúmenes de data enormes, lo que requiere muchos recursos de computación, el uso de nodos independientes no es apto debido a límites de memoria o de tiempo. Algunos de los frameworks más usados para machine learning están optimizados para usar GPUs. Existen muchos sistemas distribuidos de computación capaces de realizar entrenamiento en múltiples nodos de forma coordinada. Las aproximaciones convencionales requieren de una plataforma centralziada que recoge los datos y los distribuye entre los nodos.

Técnicas de preservación de privacidad en Machine Learning

- **Computación Multi-Grupo Segura**: Una función puede ser computada colectivamente por varios grupos sin mostrar sus propios datos. Cada grupo debe intercambiar sus salidas con otros en secreto para que puedan ser agregados. Es una de las formas más seguras de entrenar modelos de machine learnign. Por desgracia no es bueno apra entrenar modelos muy complejos o a gran escala.
- **Cifrado homomorfico**: Puede ser aplciada directamente para cifrar datos que luego son transferidos a un servidor central. Cualquier modelo puede, en teoría, ser entrenado con estos datos cifrados, obteniéndose un modelo cifrado que puede ser enviado a los clientes. El server no es capaz de descifrar ni los datos ni el modelo. Los clientes tienen acceso a sus propios datos y al modelo descifrado una vez entrenado. Este modelo tampoco es adecuado para larga escala.

Aprendizaje Federado

Es un procedimiento distribuido y colaborativo para entrenar modelos de Machine Learning sin mostrar los datos con los que se entrena. La idea de este método es mover la computación al borde, es decir, el dispositivo que obtiene los datos. Los datos no son nunca transferidos a un servidor central o a ningún almacenamiento cetnralizado, manteniendose aislados localmente. Esto mitiga vulnerabilidades relacionadas con los datos. El malchine learning es un modelo non-IID (Non-Independent identically distributed data).

Los algoritmos de aprendizaje estan descentralizados, confiando en cada nodo para realizar entrenamiento parcial del modelo. Cada nodo computa actualizaciones del modelo parcial con sus propios datos, intercambiando los parámetros con otros grupos. Un servidor central suele ser requerido para coordinar el entrenamiento, añadiendo los resultados del entrenamiento para calcular el modelo global. Una vez entrenado, el modelo global se distribuye a cada nodo para que lo pueda usar para realizar predicciones o más iteraciones del modelo.

Problemas de privacidad

Los datos no salen del dispositivo del usuario, por lo que el acceso directo por parte de terceras partes no es posible, pero sigue siendo posible la mala utilización de los datos y los modelos de Machine Learning pueden contener información sensible sobre los datos, siendo posible realizar ataques de inferencia.

Ataques de inferencia en aprendizaje federado

- Ataques de inversión de modelo
- Ataques de inferencia de miembros
- Tipos de objetivo:
  - Ataque de Caja negra (pasivo): El objetivo es el modelo final ya entrenado
  - Ataque de Caja blanca (activo): Monitoriza los cambios del modelo en cada ronda del entrenamiento
- Tipo de atacante:
  - Cliente: Puede inspeccionar las versiones consecutivas del modelo global sin interferir con el procedimiento
  - Coordinación del lado del servidor: Inspeciona las actualizaciones parciales del modelo enviada por los clientes.

Ataques de envenenamiento en Aprendizaje Federado

El adversario puede alimentar el servidor de coordinación con actualizaciones del modelo envenenadas. Normalmente se hace del lado del cliente, donde el adversario controla una fracción de los nodos participantes. El objetivo puede ser realizar un random attack apra corromper el modelo o un ataque de reemplazo, donde se inyecta una backdoor oculta dentro del modelo de forma que pueda coexistir con el modelo manteniendo el alto rendimiento.

Mecanismos de Preservación de la Privacidad

- Del lado del **servidor**:
  - **Prevención de ataques de inferencia**: Agreación segura con SMC (Secure Multiparty Computation)
  - **Prevención de ataques de evenenamiento**: Se usan técnicas de detección de anomalías como explorar los participantes del entrenamiento de los datos orealizar actualizaciones de los parámetros de inspección de los modelos.
- Del lado de los **nodos**:
  - **Prevención de ataques de inferencia**: Se aplica Agregación segura con SMC y técnicas de cifrado homomórfico. También se pueden aplicar mecanismos de privacidad diferencial para ofuscar parcialmente las actualizaciones de los modelos.

Ataques contra el Machine Learning

Existen varios tipos de ataques que se suelen realizar contra modelos de Machine Learning:

- Inferencia sobre miembros de la población:
  - Divulgación estadística
  - Inversión del modelo
  - Inferencia de representativos de una clase
- Inferencia sobre miembros del dataset de entrenamiento
  - Inferencia de membresía
  - Inferencia de propiedad
- Inferencia sobre parámetros del modelo
  - Extracción del modelo
  - Robo de funcionalidad

Ataques de inferencia sobre miembros

Tratan de determinar si se ha usado cierto input para entrenar el modelo. Basado en el comportamiento del modelo sobre datos de entrenamiento y datos ocultos.

- Ataques de caja negra: Asume el conocimiento de la salida de la predicción del modelo
- Ataques de caja blanca: Accede a parámetros del modelo y gradientes
- Ataques contra Modelos generativos: Recoge información asobre entrenamiento usando el conocimiento de los componentes generadores de datos.
- Ataque contra prendizaje federado: Un participante trata de inferir si un registro forma parte del set de entrenamiento de un participante específico o cualquier participante.
- Filtración de la cantidad de miembtros a través de las salidas de predicción.
- Problema de la inferencia de miembros. Usando un shadwo training se puede crear un sahdow model que imita el comportamiento del modelo.
  - El atacante no tiene datos para entrenar ni estradísticas sobre su distribución.
  - General datos sintéticos usando el modelo objetivo

Ataque de reconstrucción

Se trata de recrear los ejemplos de entrenamiento y sus etiquetas. La reconstrucción puede ser parcial o completa. Dadas las etiquetas de salida y conocimiento parcial sobre características se puede intentar recuperar características sensibles o toda la muestra de datos.

- Inversión del modelo:
  - Un alto nivel de error de generalización puede resultar en una mayor probabilidad de inferir atributos de los datos. Un poder predictivo mñas alto es mñas susceptible a ataques de reconstrucción.
  - El ataque se implementa de la siguiente forma:
    - El adversario tiene axceso al modelo y la salida del modelo para un ejemplo específico.
    - El ataque se basa en estimar los valores de características sensibles dados los

valores de características no sensibles y las etiquetas de salida.
- La inversión del modelo produce las características medias que mejor caracterica la salida de una clase. No construye un número específico de miembros del dataset de entrenamiento. No determina si una entrada específica fue usada para entrenar el modelo.

Ataque de inferencia de propiedades

La capacidad para extraer propiedades del dataset no codificadas como características o no correlacionadas con la tarea de aprendizaje. Este tipo de ataque tiene implicaciones de privacidad. Puede permitir a un atacante crear modelos similares. Puede ser usado para detectar vulnerabilidades en un sistema. Es posible de realizar incluso en modelos bien generalizados.

Ataque de extracción del modelo

El adversario trata de extraer información y potencialmente reconstruir el modelo. Crea un modelo sustituto que se comporta de forma similar al modelo atacado. El adversario quiere ser lo más eficiente posible.

- Task Accuracy Extraction: Para igualar la exactud del modelo objetivo, se usand atos con una distribución relacionada con los datos de aprendizaje.
- Fidelity extraction: Para hacer coincidir un set de puntos de entrada no necesariamente relacionado con la tarea de aprendizaje se crea una falsificación llamada Extracción de funcionalidad.

No es encesario saber la arquitectura del modelo bajo ataque si el modelo sustituto tiene la misma o mayor complejidad.

Técnicas de defensa en Machine Learning

Las técnicas de privacidad diferencial pueden resistir ataques de inferencia de membresía añadiendo rudio en los datos de entrada, interaciones del algoritmo de machine learning y en la salida del algoritmo.

- **Perturbación de la entrada**: Tras el enetranamiento en datos saneados, la salida será privadamente diferencial. Requiere la adición de ruido en los datos de entrada ya que estos datos suelen tener mayor sensibilidad.
- **Perturbación del algoritmo**: Aplicado a modelos de Machine learning que necesitan muchas iteraciones o pasos. Requiere un diseño específico para diferentes algoritmos de machine learning. Con los mismos recursos de privacidad diferencial suele producir menos ruido. Los valires intermedios en el etrenamiento suielen tenerr menos sensibilidad.
- **Perturbación del objetivo**: Se añade ruido a la función objetivo del algoritmo de aprendizaje. La mayoría de los mecanismos de perturbación asumen un espacio acotado. Si el espacio de muestra está acotado, el valor de cada muestra será truncado en la fase de preprocesado.

- **Perturbación de la salida**: Se usa un algoritmo no privado de aprendizaje y después se añade ruido al modelo generado. Normalmente se aplica sobre modelos que producen estadísticas complejas. No apto para muchos de los algoritmos supervisados que requieren interacturar con datos de prueba muchas veces.

Generación de datos sintéticos que preservan la privacidad

From:
https://www.knoppia.net/ - **Knoppia**

Permanent link:
https://www.knoppia.net/doku.php?id=pan:machine_learning_privacy_v2&rev=1767826826

Last update: **2026/01/07 23:00**