

Ampliación de las GNN

José Luis Rodríguez, Gonzalo Martínez y Alexandre Muñoz

December 2023

1 Introducción

GraphSAGE es un método inductivo que emplea las características presentes en los nodos para generar node embeddings sobre datos ajenos al modelo. Para ello, proponen entrenar una función que genera los embeddings muestreando y agregando características de la vecindad local de un nodo.

1.1 Arquitectura de GraphSAGE

La arquitectura GraphSAGE es un modelo de redes neuronales sobre grafos que se utiliza para aprender representaciones de nodos en grafos. A diferencia de otros enfoques que requieren el acceso a la estructura global del grafo durante la inferencia, GraphSAGE utiliza un muestreo de nodos y agregación para generar representaciones de nodos que capturan información local y global. Esto permite que GraphSAGE sea escalable a grafos grandes y que pueda aprender representaciones de nodos en grafos de manera eficiente.

Su arquitectura consta de las siguientes partes:

1. **Entrada del Grafo:**

GraphSAGE opera en un grafo dirigido o no dirigido con nodos y bordes. Cada nodo representa una entidad, y los bordes representan relaciones entre entidades.

2. **Muestreo de Vecinos:**

Para cada nodo, se selecciona un conjunto de vecinos como muestras locales para capturar información del entorno del nodo.

3. **Embedding Inicial:**

Cada nodo y sus vecinos tienen asignado un vector de características inicial, como representaciones one-hot o incrustaciones pre-entrenadas.

4. **Capas Agregadas:**

GraphSAGE utiliza capas de agregación para combinar la información de nodos y vecinos. Cada capa toma incrustaciones de un nodo y sus vecinos, produciendo una nueva incrustación para el nodo.

5. **Agregación por Media o Agregación por Pooling:**

La agregación implica combinar incrustaciones de vecinos para producir una nueva incrustación para el nodo. Puede ser por media o pooling (por ejemplo, max pooling).

6. **Múltiples Capas:**

GraphSAGE puede tener múltiples capas de agregación para capturar información contextual a diferentes niveles de proximidad en el grafo.

7. **Salida:**

La salida final es una incrustación que captura información contextual y estructural del grafo.

8. **Tareas Específicas:**

La salida se utiliza para diversas tareas en grafos, como clasificación de nodos, regresión, clustering, etc. Se añade una capa específica para la tarea deseada en la parte superior de las capas de GraphSAGE.

GraphSAGE es eficiente y escalable, ya que realiza muestreo de vecinos para evitar procesar todo el grafo, lo que lo hace aplicable a grafos grandes. Además, al incorporar información contextual a través de capas de agregación, puede aprender representaciones ricas y significativas para nodos en el grafo.

1.2 Descripción de cómo Pinterest adapta GraphSAGE para sus aplicación

Usan PinSage, una red convolucional de gráficos de recorrido aleatorio capaz de aprender incrustaciones de nodos en gráficos a escala web que contienen miles de millones de objetos.

PinSage, opera en un gráfico masivo con tres mil millones de nodos y 18 mil millones de aristas, un gráfico que es 10 000 veces más grande que las aplicaciones típicas de GCN. PinSage aprovecha varios conocimientos clave para mejorar drásticamente la escalabilidad de los GCN.

1. **Convoluciones sobre la marcha:**

PinSage realiza convoluciones localizadas eficientes al muestrear la vecindad alrededor de un nodo y construir dinámicamente un gráfico de cálculo. Estos gráficos de cálculo contruidos dinámicamente especifican cómo realizar una convolución localizada alrededor de un nodo en particular y alivian la necesidad de operar en todo el gráfico durante el entrenamiento.

2. **Construir convoluciones mediante paseos aleatorios**

PinSage define vecindarios basados en la importancia simulando caminatas aleatorias y seleccionando a los vecinos con el mayor número de visitas. Las ventajas de esto son dos:

–Primero, permite a sus agregadores tener en cuenta la importancia de los vecinos al agregar las representaciones vectoriales de los vecinos. Se referimos a este nuevo enfoque como agrupación de importancia.

–En segundo lugar, seleccionar una cantidad fija de nodos para agregar les permite controlar la huella de memoria del algoritmo durante el entrenamiento.

Su enfoque propuesto basado en caminatas aleatorias conduce a una ganancia de rendimiento del 46/100 sobre el método tradicional de vecindario de gráficos K-hop en nuestras métricas de evaluación fuera de línea.

3. Inferencia eficiente de MapReduce

En cada paso de agregación, se utiliza la función de mapeo para proyectar todos los nodos al espacio latente sin realizar cálculos duplicados. Luego, se fusionan estas proyecciones para ser enviadas a los nodos correspondientes de nivel superior en la jerarquía. Finalmente, se realiza una reducción para llevar a cabo la agregación y obtener las incrustaciones de los nodos de nivel superior. La eficiente inferencia basada en MapReduce permite la generación de incrustaciones para miles de millones de nodos en unas pocas horas, incluso en un clúster compuesto por solo unas pocas centenas de instancias. Este enfoque facilita el procesamiento escalable de grandes conjuntos de datos en entornos distribuidos, destacando la capacidad de generar representaciones latentes de nodos de manera rápida y efectiva.

4. Inferencia eficiente de MapReduce

El rendimiento de PinSage se evaluó en comparación con varias líneas base de aprendizaje profundo basadas en contenido que generan incrustaciones de Pines. Estas líneas base incluyen:

Incrustaciones Visuales (Visual): Utiliza los vecinos más cercanos de incrustaciones visuales profundas para generar recomendaciones.

Incrustaciones de Anotaciones (Annot.): Genera recomendaciones basándose en los vecinos más cercanos en términos de incrustaciones de anotaciones.

Incrustaciones Combinadas (Combinadas): Recomienda en función de la concatenación de incrustaciones visuales y de anotaciones anteriores. Utiliza un perceptrón multicapa de 2 capas para calcular incrustaciones que capturan características visuales y de anotaciones. Este modelo está entrenado con exactamente la misma función de pérdida y datos que PinSage.

Estas líneas base sirven como puntos de referencia para evaluar la efectividad de PinSage en la generación de incrustaciones y recomendaciones. La comparación se basa en métricas específicas, y la eficacia relativa de PinSage se determina en función de su capacidad para superar o igualar el rendimiento de estas líneas base en el contexto de recomendaciones de Pines.

1.3 Descripción de cómo UberEats adapta GraphSAGE para sus aplicación

En el feed principal de Uber Eats, se generan carruseles de recomendaciones tanto para restaurantes como para elementos del menú, adaptados a las preferencias de los usuarios. Durante la navegación por el menú de un restaurante, se generan recomendaciones personalizadas de artículos dentro de ese establecimiento, utilizando sistemas de recomendación capacitados en pedidos anteriores y preferencias de los usuarios.

El sistema de recomendación de Uber Eats se desglosa en dos fases principales: la generación de candidatos y el ranking personalizado. El componente de generación de candidatos tiene como objetivo producir candidatos relevantes, como platos y restaurantes, de manera escalable. Esta fase se diseñó para ser altamente escalable y permitir el filtrado previo de la creciente variedad de opciones de platos y restaurantes en la plataforma. El prefiltrado considera factores como la ubicación geográfica para asegurar que se recomienden opciones dentro del rango de entrega del usuario y que los candidatos sean pertinentes a las preferencias específicas de cada usuario.

El segundo componente, el clasificador personalizado, es un modelo de aprendizaje automático completo que clasifica los platos y restaurantes candidatos prefiltrados. Esta clasificación se basa en información contextual adicional, como el día, la hora y la ubicación actual del usuario al abrir la aplicación Uber Eats. El modelo puede aprender patrones recurrentes de pedidos, como preferencias específicas de comida en ciertos días de la semana o la elección de diferentes tipos de platos para el almuerzo y la cena.

Para mejorar aún más las recomendaciones, se adoptó un enfoque basado en grafos utilizando GraphSAGE. Se crearon dos gráficos bipartitos: uno que representa a los usuarios y los platos, y otro que representa a los usuarios y los restaurantes. Estos gráficos contienen información sobre la frecuencia con la que un usuario ordena un plato específico o en un restaurante particular.

GraphSAGE fue elegido debido a su escalabilidad y capacidad para manejar gráficos de gran tamaño. La función de agregación de GraphSAGE implica agrupación máxima o media después de una proyección. Se realizó una proyección adicional para manejar diferentes características en cada tipo de nodo (usuario, restaurante o plato). Además, se implementó una estrategia de muestreo para limitar el número de nodos considerados en el aprendizaje, permitiendo escalar el modelo a gráficos con miles de millones de nodos.

Para adaptar GraphSAGE a estos gráficos bipartitos, se introdujo una capa de proyección adicional para manejar características distintas en cada tipo de nodo. Además, se modificó el enfoque para tener en cuenta aristas ponderadas, esenciales para incorporar información sobre la cantidad de veces que un usuario ordena en un restaurante o un plato específico, así como la calificación otorgada por un usuario a un plato. Estas adaptaciones permitieron que GraphSAGE ofreciera recomendaciones más precisas y contextualmente relevantes a través de la consideración de diversas señales importantes en el proceso de recomendación de Uber Eats.

2 Ideas sobre cómo PinSAGE y la variante de UberEats podrían haber modificado GraphSAGE.

2.1 PinSAGE para Pinterest

Manejo de Grafos a Escala Web:

PinSAGE opera en un grafo masivo con miles de millones de nodos y 18 mil millones de aristas. Esta adaptación podría incluir optimizaciones específicas para lidiar con la escala web, como técnicas de paralelización y distribución de cálculos para mejorar la eficiencia en entornos de gran escala.

Convoluciones Eficientes:

PinSAGE utiliza convoluciones sobre la marcha al muestrear la vecindad alrededor de un nodo. Las adaptaciones podrían incluir mejoras en la eficiencia del muestreo y la construcción dinámica de gráficos de cálculo para acelerar el proceso de convolución.

Vecindarios Basados en Importancia:

La elección de vecindarios basados en la importancia a través de caminatas aleatorias podría haberse ajustado para mejorar aún más la captura de relaciones importantes en el contexto de Pinterest, donde la relevancia de los pines y usuarios puede variar significativamente.

Inferencia Eficiente:

Dada la enorme escala del grafo, las mejoras en la eficiencia de la inferencia, como la utilización de MapReduce, son cruciales. Las adaptaciones podrían haber incluido optimizaciones específicas para procesar grandes conjuntos de datos de manera eficiente en entornos distribuidos.

2.2 Variante de GraphSAGE para UberEats

Generación de Candidatos Escalable:

La variante de GraphSAGE para UberEats se enfoca en la generación de candidatos relevantes para platos y restaurantes de manera escalable. Podrían haber introducido técnicas adicionales de muestreo y filtrado previo para manejar la enorme variedad de opciones en su plataforma.

Clasificador Personalizado:

UberEats utiliza un clasificador personalizado basado en información contextual adicional. Las modificaciones podrían haberse centrado en la mejora de la capacidad predictiva del clasificador mediante la inclusión de características específicas del servicio de entrega de alimentos, como la disponibilidad de promociones o la velocidad de entrega.

Manejo de Grafos Bipartitos:

La adaptación de GraphSAGE para gráficos bipartitos con nodos de usuario, restaurante y plato podría haber requerido ajustes en la estrategia de muestreo y agregación para manejar eficazmente la diversidad de nodos y relaciones en el grafo.

Consideración de Aristas Ponderadas:

La introducción de aristas ponderadas en el grafo bipartito podría haber implicado ajustes en la función de agregación para tener en cuenta la importancia de las conexiones y las señales asociadas, como la frecuencia de pedidos y las calificaciones.

Estas ideas representan posibles áreas de adaptación y modificación en PinSAGE para Pinterest y la variante de GraphSAGE para UberEats, respectivamente, con el objetivo de optimizar y personalizar el rendimiento de los modelos para sus aplicaciones específicas.

Reflexiones sobre Otras Posibles Aplicaciones y Adaptaciones de GraphSAGE

Redes Sociales:

GraphSAGE podría aplicarse a redes sociales para modelar la interconexión de usuarios y sus relaciones. Sería útil para predecir conexiones futuras, identificar comunidades y personalizar recomendaciones de contenido.

Búsqueda Semántica:

En motores de búsqueda, GraphSAGE podría utilizarse para mejorar la comprensión semántica de los documentos y consultas. Al modelar la relación entre palabras y documentos, podría ofrecer resultados más precisos y contextualmente relevantes.

Bioinformática:

En biología computacional, GraphSAGE podría adaptarse para analizar redes biológicas. Ayudaría en la predicción de funciones de genes, la identificación de comunidades de proteínas y la comprensión de la interacción gen-proteína.

Fraude en Transacciones Financieras:

GraphSAGE podría ser útil para detectar patrones de fraude en redes de transacciones financieras. Modelando las relaciones entre cuentas y transacciones, podría identificar comportamientos anómalos y transacciones sospechosas.

Recomendaciones de Contenido Multimedia:

En plataformas de contenido multimedia, GraphSAGE podría mejorar las recomendaciones personalizadas. Al modelar las relaciones entre usuarios, contenido y preferencias, podría ofrecer sugerencias más precisas para películas, música o libros.

Logística y Rutas de Entrega:

Para optimizar la logística y rutas de entrega, GraphSAGE podría modelar la red de ubicaciones, almacenes y entregas. Esto ayudaría a planificar rutas eficientes y prever posibles congestiones en la cadena de suministro.

Colaboración Científica:

En entornos de colaboración científica, GraphSAGE podría modelar la red de colaboraciones entre investigadores y laboratorios. Facilitaría la identificación de expertos en un campo y la predicción de posibles colaboraciones.

Detección de Anomalías en Sistemas:

GraphSAGE podría aplicarse para la detección de anomalías en sistemas complejos, como redes de computadoras. Modelando la topología de la red, podría identificar comportamientos inusuales y posibles amenazas de seguridad.

Estas reflexiones sugieren que las posibles aplicaciones de GraphSAGE son amplias y abarcan diversos campos. La capacidad de modelar relaciones en grafos permite su adaptación a escenarios complejos donde la interconexión entre entidades es fundamental para comprender y tomar decisiones informadas.