Introducción a las GNN

Andrea Manuel Simón

November 2023

1 Teoría y Fudamentos

Las Graph Network (GNNs) son un tipo de modelo de aprendizaje profundo diseñado específicamente para procesar datos en una estructura de gráfos. Una GNN se compone de capas que operan sobre el grafo, teniendo en cuenta la información de los nodos y sus conexiones para realizar el aprendizaje. Cada capa comparte información a través de las aristas del grafo, perimitiendo que los nodos compartan información con sus vecinos.

Operaciones básicas en GNN:

-Agregación de vecinos: para cada nodo la GNN agrega la información de sus nodos vecinos. Esta operación puede ser una suma o una media ponderada.

-Transformación Nodal: la información agregada al nodo se combina con la representación original para obtener una nueva representación.

-Actualización de Nodos: la representación actualizada del nodo se utiliza para actualizar su estado en la siguiente capa de la red. Este proceso se repite a través de múltiples capas.

Las capas de los GNN se pueden clasificar en capa de agregación y capa de actualización. La capa de agregacón calcula la información agregada de los vecinos de un nodo. Por otro lado, la capa de actualización utiliza la información agregadad para actualizar la representación del nodo.

Una propiedad interesante de las GNN es su capacidad para generar nuevos grafos que no han sido vistos durante el entrenamiento, haciendo que sean efectivas para problemas en los que la estructura de la red varia.
Estas se utilizan para la clasificación de nodos, predicción de enlaces, recomendación de productos y análisis de redes sociales.

2 Implementación

Vamos a intentar rediseñar la capa convolucional del gráfico desde cero.
Sabemos que en las redes neuronales las capas lineales aplican una transformación lineal a los datos entrantes, transformando las características de entrada x en vectores ocultos h usando una matriz de peso W. Si ignoramos los sesgos podemos escribir:

$$h = Wx \tag{1}$$

Con esto tenemos acceso a la información entre nodos.
Se puede dar el caso de que en la mayoría de las redes nos encontramos con homofilia de red, es cuando hacemos la hipótesis de que es más probable que los nodos similares estén conectados entre sí que los diferentes.

Vamos a realzar una convolución o agrregación de vecindad para enriquecer la representación de nodos mediante la agregación de sus características con las de los vecinos. Ñi, es el vecindatio del nodo i incluyendole. Nuestra matriz de

$$h_i = \sum_{j \in \tilde{\mathcal{N}}_i} \mathbf{W} x_j$$

peso, W, es única y se comparte entre todos los nodos.

A pesar de todo, nos surge un problema, los nodos no tienen un número fijo de vecinos como los pixeles. Puede dar el caso que un nodo tenga solo un vecino y otro 500, donde agregaríamos los 500 valores en lugar de solo uno haciendo que h sea mucho más grande para el nodo con los 500 vecinos.
Esto no tiene sentido, dado que los nodos siempre deben de tener un rango de valores similares. Para solucionar esto podemos normalizar el resultado en función del número de conexiones, en otras palabras ajustaríamos el grado. Tras

$$h_i = \frac{1}{\deg(i)} \sum_{j \in \tilde{\mathcal{N}}_i} \mathbf{W} x_j$$

realizarlo podemos ver que los nodos con muchos vecinos se propagan de una forma más sencilla que los nodos más aislados, para constrarestarlo daremos más peso a los nodos con menos vecinos. Cuando i y j tienenn el mismo número

$$h_i = \sum_{j \in \tilde{\mathcal{N}}_i} \frac{1}{\sqrt{\deg(i)}\sqrt{\deg(j)}} \mathbf{W} x_j$$

de vecionos es quivalente a nuestra propia capa.

3 Conslusiones de la implementación

Aquí mostramos las conslusiones del código que se encuentra en el repositorio.

Figure 1:
En esta primera imagen podemos ver el resultado del entrenamiento del modelo, en el que vemos una precisión del 100

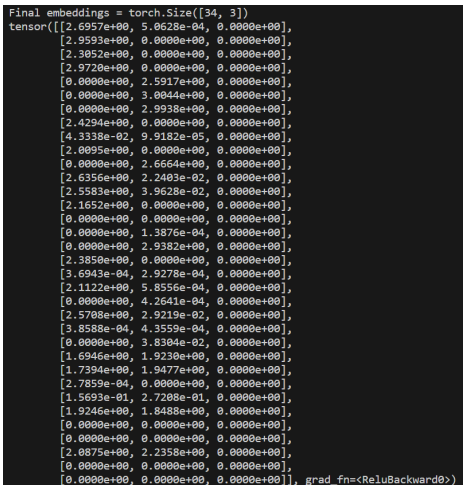

Figure 2:
Al agregar características de los nodos vecinos, la GNN aprende una representación vectorial de cada nodo en la red. En nuestro modelo, la capa final solo aprende a usar estas representaciones para producir las mejores clasificaciones. Sin embargo, las incrustaciones son los productos reales de las GNN. Esto se puede ver en la imagen del tensor.

4 Conslusiones

Las GNN y GCN han demostrado ser herramientas poderosas en una variedad de contextos y problemas. Se pueden aplicar las GNN en redes sociales, donde estos son esenciales para el análisis de comunidades, identificación de influenciadores y recomendaciones personalizadas. Otro ejemplo es en las recomendaciones y filtrado, son aplicables para modelar la iteracción entre usuarios y elementos superando las limitaciones de métodos tradicionales. Las GCN se pueden aplicar en la clasificación y segmentación de imágenes al aprovechar la estructura de los objetos en la escena. También en el procesamiento de señales y datos temporales, donde la información está organizada en forma de grafo.

En conclusión, las GNN y GCN han demostrado ser eficaces y versátiles en diversos campos, aunque la interpretación de sus decisiones puede ser un gran desafio. Además, tenemos que tener en cuenta la eficiencia computacional y la escalabilidad en grafos grandes. De esta manera, su relevancia seguirá creciendo a medida que se desarrollen nuevas técnicas y se apliquen en contextos

emergentes. La comprensión de estas redes y su aplicación efectiva requiere un enfoque cuidadoso y adaptativo según el problema y el conjunto de datos específicos.