Análisis de topología de grafos para encontrar contrafractuales de modelos de caja negra

Kevin Manzano Rodríguez

11 de diciembre de 2024

Universidad de La Habana
Facultad de Matemática y Computación
Tesis de Licenciatura

Kevin Manzano Rodríguez *Trabajo realizado bajo la dirección del Profesor*
Rodrigo García Gómez

Resumen

Propuesta de heurística para mejorar la elección de aristas para perturbar en el minimizador.

Índice general

Capítulo 1

Introducción

Sea $G = (V, E)$ un grafo general. Se cuenta con un explainer denominado `generator-minimizer` que opera en dos pasos. En el primero se genera un contrafractual (un grafo G' creado a partir de G que dado un modelo cambia la predicción con respecto a G). Luego este se pasa como entrada al minimizador, para encontrar uno más pequeño. En el segundo paso, se utiliza una heurística la cual tiene un componente aleatorio para seleccionar aristas a perturbar (cambiar de estado, si la arista está en G se retira, y si no está se agrega), se busca mejorar esta componente en este documento dándole un sentido a la elección para hacerla más efectiva.

Capítulo 2

Capítulo 2

En muchos modelos de forma general se observa que existen componentes del grafo que parecen tener un peso grande en la predicción final del mismo, dígase nodos muy conectados entre sí, aristas que forman ciclos, aristas que al quitarlas se pierde la conexión del grafo ect. Dado que no sabemos con que modelo estamos tratando vamos a realizar un preprocesamiento para según el modelo en cuestión, él mismo le de sentido a las relaciones.
Definamos la función $f : E \times E \to \mathbf{N}$, tal que $f(e_1, e_2) = r$, esta toma dos aristas e_1, e_2 de nuestro grafo G y devuelve un natural r que exprese que tan 'relacionadas' están e_1, e_2 según el modelo en cuestión. (Notese que es simétrica y $f(e, e) = \infty$)

Intuitivamente una idea para calcular el valor de f en un par de aristas e_1, e_2, sería fijemos e_1, e_2 en un estado (perturbar ambas, no perturbar ambas, perturbar la primera, perturbar la segunda) e ir moviendose por todos los posibles grafos para cada estado del par de aristas fijado, luego pasarlo al modelo y comparar las predicciones. Luego de obtenidos los resultados para un estado fijado del par de aristas, tendremos 4 cantidades a, b, c, d en el mismo orden mencionado anteriormente, entonces tendría sentido decir que si $a + b > c + d$ es porque aparentemente el estado de las aristas fijadas si se perturban a la vez o no se perturban a la vez hace que el modelo cambie de predicción. Pero sería más provechoso diferenciar en todos los casos pues lo que nos interesa es construir un conjunto de aristas a perturbar que mejore la heurística por tanto, definamos P como el conjunto de aristas a perturbar en el paso k de la heurística, digamos que una de las cantidades (a, b, c, d) es buena si en al menos la mitad de los casos cambia la predicción.

1. Si $e_1 \in P \wedge e_2 \in P$
   a) Si a es buena lo dejamos asi.
   b) Si a no es buena vemos cual de las cantidades es la más grande y cambiamos el estado de e_1 y/o e_2.

   .

2. Si $e_1 \in P \wedge e_2 \notin P$
   a) Si c es buena lo dejamos asi.

   b) Si c no es buena vemos cual de las cantidades es la más grande y cambiamos el estado de e_1 y/o e_2.

3. Si $e_1 \notin P \wedge e_2 \in P$

   a) Si d es buena lo dejamos asi.

   b) Si d no es buena vemos cual de las cantidades es la más grande y cambiamos el estado de e_1 y/o e_2.

4. Si $e_1 \notin P \wedge e_2 \notin P$

   a) Si b es buena lo dejamos asi.

   b) Si b no es buena vemos cual de las cantidades es la más grande y cambiamos el estado de e_1 y/o e_2.

Notese el patrón, de que dado el estado del par y las cantidades la elección es trivial.

Pero evidentemente esta vía no es factible pues si pudieramos realizar todo este computo, directamente el algoritmo pudiera iterar por todos los grafos buscando el menor contrafractual, entonces abstraigamos todo este computo en algunas pruebas de forma aleatoria, o sea digamos que nuestro proceso para calcular un par de aristas que pasaba por todo los grafos con esas aristas fijadas, ahora pasará por una cantidad aleatoria de grafos elegidos de forma aleatoria, para tener una idea de cuanto influiria esto de forma general.

Capítulo 3

Capítulo 3

Supongamos que tenemos todas las categorias existentes en nuestro dataset $C_1, C_2, ..., C_m$ y queremos generar un contrafractual para el grafo G con el modelo M, si tenemos al menos una categoría diferente C_i donde esta es a la que pertenece G, un contrafractual válido para G con M sería cualquier grafo que pertenece a C_j con $j \neq i$. También sería bueno tener grafos cercanos de forma general a todos los grafos de la entrada, por lo que vamos a construir una idea de cercania.

Dado una categoría del grafo C, formemos el subconjunto de los grafos del dataset que pertenecen a C (realizando una llamada al oráculo podemos saber a que categoría pertenece), nos concierne encontrar un grafo especial intuitivamente cercano a todos los grafos de dicha categoría, si se logra modelar los grafos como puntos en un espacio, entonces el problema se reduciría a encontrar el punto que minimiza la distancia a todos los puntos de la categoría.
La modelación del espacio queda por concretar, pero se pudiera pensar en un espacio en el cual las características son las aristas del grafo y la distancia entre dos puntos es la cantidad de aristas que difieren entre ellos.
Dado que queremos encontrar el x que minimiza la distancia de los n grafos de la categoría C hasta él, $\sum_{i=1}^{n} \sqrt{\sum_{j=1}^{m}(x_j - y_{ij})^2}$ donde m es la dimensión del espacio dada por $E * (E-1)/2$ del mayor grafo, x_j la característica $j - esima$ de x, y y_{ij} la característica $j - esima$ del $i - esimo$ grafo; pero notemos que esto es análogo a $\sum_{i=1}^{n}\sum_{j=1}^{m}(x_j - y_{ij})^2$. Por tanto analizando la derivada en una componente x_j de la función objetivo, se tiene que
$\frac{\partial}{\partial x_j}\sum_{i=1}^{n}\sum_{j=1}^{m}(x_j - y_{ij})^2 = 2\sum_{i=1}^{n}(x_j - y_{ij}) = 0$ por tanto $x_j = \frac{1}{n}\sum_{i=1}^{n} y_{ij}$, o sea que el punto que minimiza la distancia es el promedio de las características de los grafos de la categoría.
Luego podemos tener un grafo 'cercano' a cualquier grafo en cada una de las categorías, por lo que obtener un contrafractual consistirá en iterar por todas las categorías diferentes a la que pertenece el grafo en cuestión y devolver el más cercano a él.