

Sándor Zsuzsanna

Gráfkonvolúciós neuráliahálózatok

Önálló projekt beszámoló, I. félév

1. Gráfkonvolúció elméleti háttere

A fogalmak jelentős részét Harsányi Benedek Gráf konvolúciók és alkalmazásai [1] című szakdolgozatából értettem meg.

A gépi tanulási klasszifikációs feladatoknál adott egy tanító adathalmaz $\{x_i, y_i : i = 1, \dots, k\}$. Ez alapján találnunk kell egy f_θ függvényt, hogy $\forall i : f_\theta(y_i) \approx x_i$. A neurális háló feladata, hogy a $\theta \in \mathbb{R}^p$ paramétereket beállítva megvalósítsa ezt az approximációt. A gráfkonvolúciós háló esetén az y_i magyarázó változó egy gráf csúcsain és élein felvett értékek lehetnek.

1.1. Definíció ([5]). Legyen $G = (V, E, W)$ gráf, $W : V \times V \rightarrow \mathbb{R}$ súlyfüggvény, úgy hogy $W(i, j) = w_{ij}$, ha $(i, j) \in E$, különben 0, $w_i = \sum_{j \in N(i)} w_{ij}$. Ekkor a normalizált Laplacian-mátrix

$$(L_n)_{ij} := \begin{cases} 1, & \text{ha } i = j, \\ \frac{-w_{ij}}{\sqrt{w_i w_j}}, & \text{ha } (i, j) \in E, \\ 0 & \text{ha } (i, j) \notin E. \end{cases}$$

Az L_n hez tartozó sajátértékeket $0 \leq \lambda_1 \leq \dots \leq \lambda_n$ a gráf (normalizált Laplace-)spektrumának nevezzük, és sorozatukat $\sigma(L_n)$ -nel jelöljük.

A normalizált Laplace-mátrix szimmetriája miatt SVD felbontása segítségével előáll $L_n = \Phi \Lambda \Phi^T$ alakban, ahol $\Phi = [\phi_1, \dots, \phi_n]$ a sajátvektorok mátrixa és $\Lambda = \text{diag}([\lambda_1, \dots, \lambda_n])$ pedig a sajátértékekből alkotott diagonális mátrix.

A konvolúció fogalmához szükségünk van a Fourier-transzformáltakra, mivel a konvolúció eredeti formáját nehéz lenne közvetlenül átültetni a gráfok függvényeire. Két függvény konvolúciója kifejezhető e két függvény Fourier-transzformáltjának segítségével: $(f * g) = \mathcal{F}^{-1}\{\mathcal{F}\{f\} \cdot \mathcal{F}\{g\}\}$.

Ezzel a formulával gráfokon értelmezett függvényekre is általánosítható a konvolúció fogalma.

1.2. Definíció ([5]). A $G = (V, E)$ gráf csúcsain értelmezett $x : V \rightarrow \mathbb{R}$ függvény Fourier-transzformáltja

$$\hat{x}(\lambda_j) = \langle x, \phi_j \rangle = \sum_{i=1}^n x(i) \phi_j(i).$$

Az inverz Fourier-transzformált pedig

$$x(i) = \sum_{j=1}^n \hat{x}(\lambda_j) \phi_j(i).$$

1.3. Definíció (Gráfkonvolúció [5]). A $G = (V, E, W)$ gráf csúcsain értelmezett $x, g : V \rightarrow \mathbb{R}$ függvények konvolúcióján a következő kifejezést értjük

$$(x * g) = \sum_{j=1}^n \hat{x}(\lambda_j) \hat{g}(\lambda_j) \phi_j(i).$$

Mátrixszorzással felírva, illetve \circ jelölve a vektorok koordinátánkénti szorzatás, a konvolúciós formulával analóg képletet kapunk a konvolúció függvényére $(x * g) : V \rightarrow \mathbb{R}$, $(x * g) = \Phi((\Phi^T x) \circ (\Phi^T g))$. Itt a g függvényt filter operációnak is nevezzük.

Feltéve, hogy a g függvény sorbafejthető, a konvolúcióra igaz, hogy $(x * g) = \Phi((\Phi^T x) \circ (\Phi^T g)) = \Phi((\Phi^T x) \circ (\hat{g})) = \Phi \text{diag}(\hat{g}(\lambda_1), \dots, \hat{g}(\lambda_n)) \Phi^T x = \Phi \hat{g}(\Lambda) \Phi^T x = \hat{g}(L_n)x$.

1.4. Definíció (Gráfkonvolúciós réteg [5]). *Input:* $x_1, \dots, x_q \in \mathbb{R}^{|V|}$, *output:* $\hat{x}_1, \dots, \hat{x}_{q'} \in \mathbb{R}^{|V|}$, $\xi : \mathbb{R} \rightarrow \mathbb{R}$ a csúcsonként alkalmazott aktivációs függvény. Ekkor konvolúciós rétegnek nevezzük az alábbi hozzárendelést:

$$\hat{x}_j = \xi \left(\sum_{i=1}^q g_j(L_n)x_i \right) \quad j \in \{1, \dots, q'\}$$

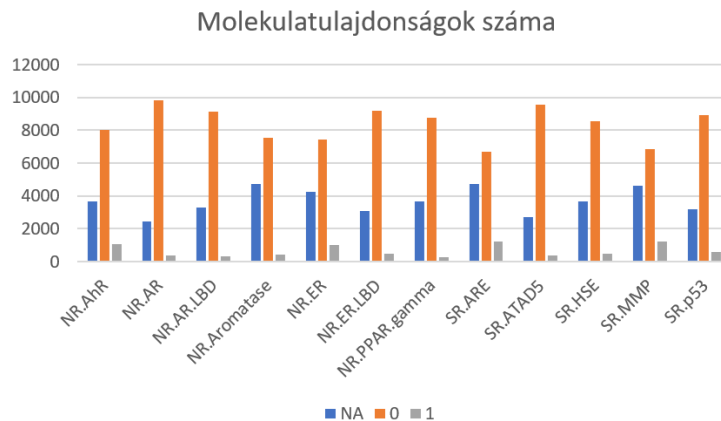
A $\hat{g} : \{\lambda_1, \dots, \lambda_n\} \rightarrow \mathbb{R}$ függvény által definiált gráf filternek nevezzük a $\sum_{i=1}^n \phi_i \hat{g}(\lambda_i) \phi_i^T = \Phi \hat{g}(\Lambda) \Phi^T = \hat{g}(L_n)$ kifejezést. Ez lehet k -ad fokú, $\Theta \in \mathbb{R}^{k+1}$ paraméterű polinomiális filter, ha $\hat{g}(L_n) = \sum_{i=0}^k \theta_i L_n^i$ alakú. Ezek a filterek úgynevezett lokális filterek, mivel egy csúc értékét csak a tőle legfeljebb k távolságban lévő csúcsok fogják meghatározni. Filterek készítésére több módszer is ismert, mint a Csebisev-polinomokkal definiált filter, Cayley-filter, ARMA filter vagy a Spline-okat felhasználó filter. Ez utóbbi filter nem a spektrális gráfelméleten alapul.

A gráfkonvolúciókat többek között használják képek feldolgozására a pixeleket rácsgráfba rendezve, és egyéb gráfokkal modellezhető problémák megoldására, mint a fehérjék, molekulák osztályozása.

2. A Tox21 adathalmaz

A gyógyszerfejlesztés hatékonysága is növelhető gépi tanulás segítségével. Arra a feltevésre építve, hogy a molekula szerkezete meghatározza kémiai tulajdonságait, a molekula szerkezetét megfelelően reprezentálva megjósolható, hogy több ezer vegyület közül melyek milyen hatással lehetnek a szervezetre. Ezáltal kiválaszthatók azok a molekulák, amelyekkel érdemes lehet kutatásokat folytatni, így csökkenthető a laboratóriumi kísérletek száma.

A Tox21 Data Challenge [3] nevű adatbázis is az ilyen irányú törekvéseket ösztönzi 12707 természetes vegyületet és gyógyszert felsorakoztató adathalmazával, melyek 12 tulajdonság meglétével, hiányával vagy ismeretlenségével vannak jellemezve. A gépi tanulási modell az adathalmazból kinyert információk alapján, később csupán a molekula szerkezetéből meg tudja állapítani a vegyület jellemzőit.



Az adatokat a Pytorch Geometric könyvtár [2] használatával tervezem feldolgozni, amit kifejezetten a struktúrált adatok gráfkonvolúciókkal való kezelésére fejlesztettek. A feladat GPU-igénye miatt Google Colabban vagy az ELTE szervereiről fogom futtatni a modelleket.

A Python `torch_geometric` TUDataset [4] nevű könyvtárából is letölthető Tox21 adathalmazt, ami tartalmazza a molekulagráf éllistáját, valamint külön fájlokban az élek, csúcsok és a gráfok címkéit.

Ebben a félévben ismerkedtem meg a deep learning alapjaival is: neurális háló, backpropagation, aktivációs függvények, regularizációs megoldások, konvolúciós háló és előtanított háló. Ehhez a múltfélelvi deep learning kurzus anyaga szolgált segítségül. A továbbiakban a megszerzett ismeretekre építve szeretnék minél jobb teljesítményű gráfkonvolúciós modellt alkotni a Tox21 adathalmazhoz.

Hivatkozások

- [1] Harsányi Benedek. Gráf konvolúciós hálózatok és alkalmazásaik. *szakdolgozat*, 2021.
- [2] Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.
- [3] Andreas Mayr, Günter Klambauer, Thomas Unterthiner, and Sepp Hochreiter. Deeptox: toxicity prediction using deep learning. *Frontiers in Environmental Science*, 3:80, 2016.
- [4] Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. TU-Dataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*, 2020.
- [5] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.