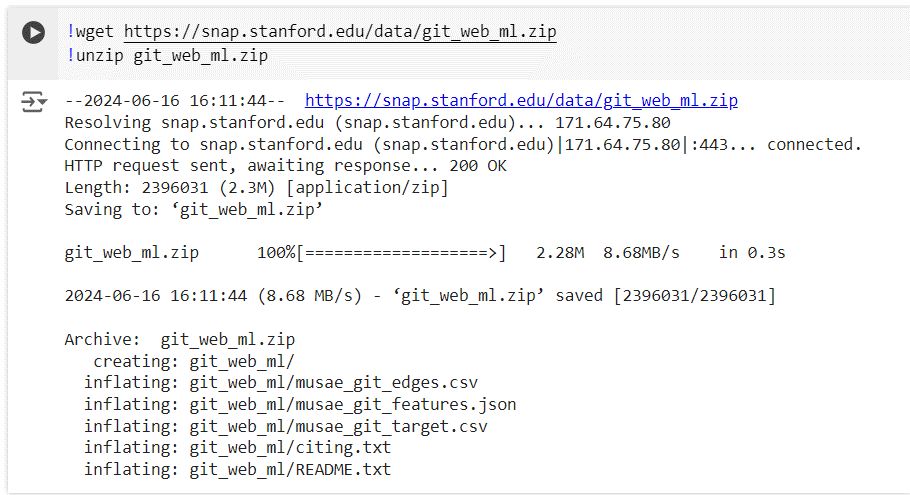
**تمرین ششم درس داده کاوی محمدباربد امیرمزلقانی – 810102348**

**بخش عملی(فایل نوتبوک در ضمیمه موجود است)**

**الف)** در این بخش ابتدا به کمک دستور wget دیتاست را دانلود میکنیم و آن را آنزیپ میکنیم :

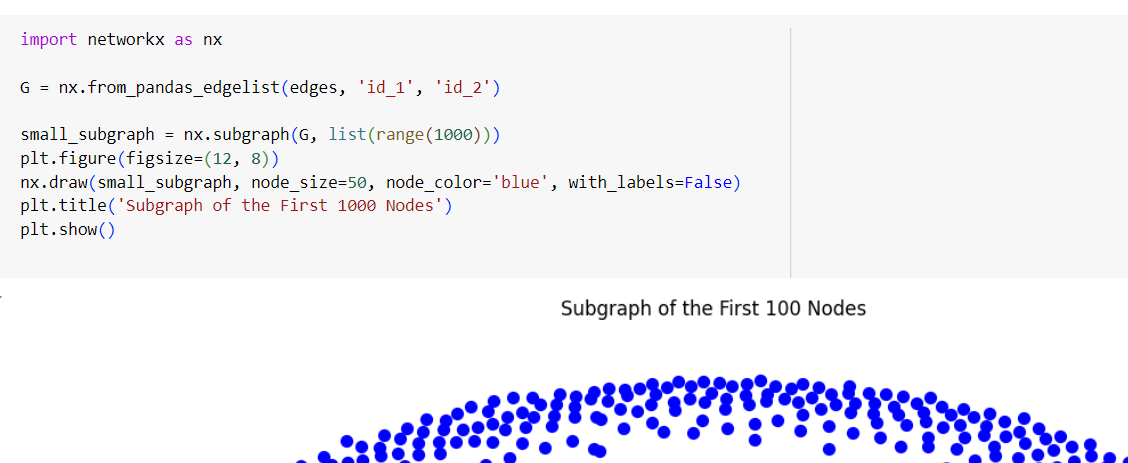
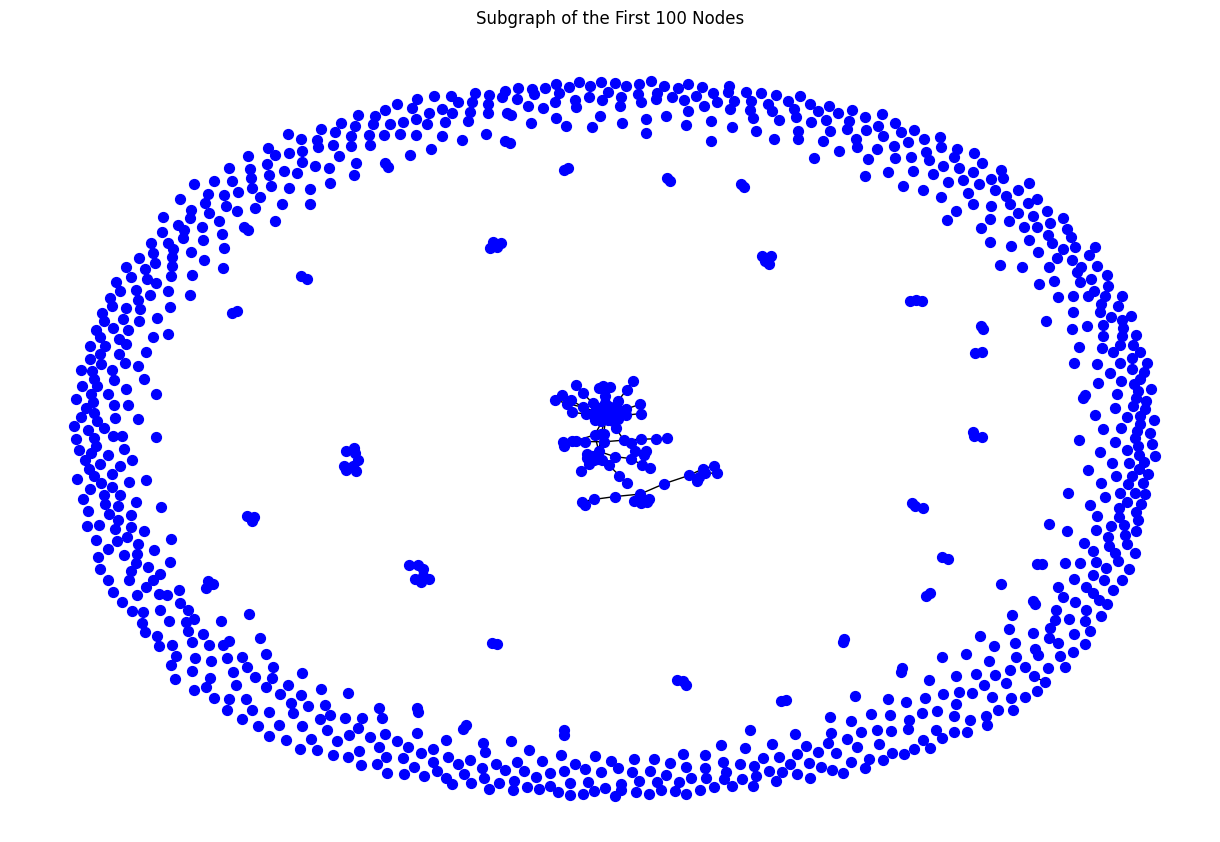


در ادامه باید به بررسی فایل ها بپردازیم و ببینیم هر کدام شامل چه اطلاعاتی هستند، در همین راستا دو فایل csv را باز میکنیم و طبق پیش بینی موارد خواسته شده هستند که در ادامه توضیح میدهم :



ابتدا فایل edges که شامل تمامی یال های موجود میباشد، این فایل دو ستون که شامل راس هایی هستند که آن یال به هم متصل میکند، فایل دوم که بیانگر متغیر هدف ماست و به ازای هر آیدی ویژگی نهایی که میخواهیم پیش بینی ( مربوط به ml) کنیم را شامل میشود.(همچنین نام کاربر هم موجود است که در اینجا زیاد از آن استفاده نمیکنیم)

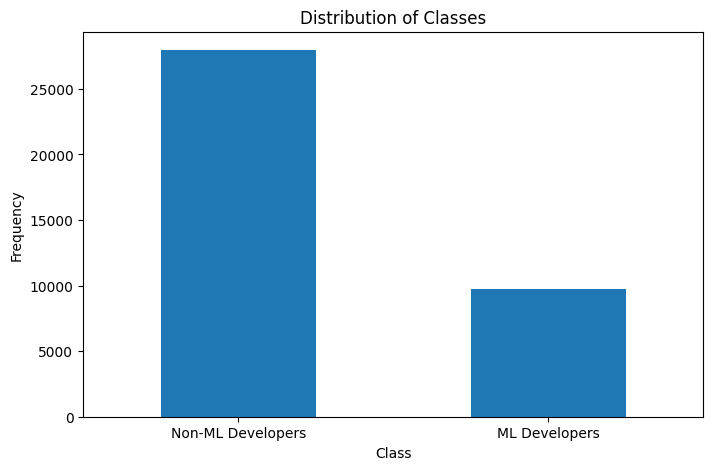
**ب)** در این قسمت لازم بود تا یک سری visualization انجام بدهیم، تا درک بهتری از دیتاست داشته باشیم همچنین بالانس بودن را لازم بود در این قسمت چک کنیم، ابتدا سعی میکنیم گراف را برای هزار نود اول رسم کنیم که خروجی به صورت زیر میشود (طبیعتا کامل نیست چون فقط هزار تای اول بودن )(برای نمایش هم از کتابخانه networkx استفاده میکنیم :

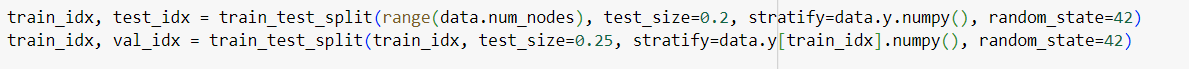
در ادامه به سراغ بالانس بودن میرویم و از نمودار میله ای برای این سنجش استفاده میکنیم :



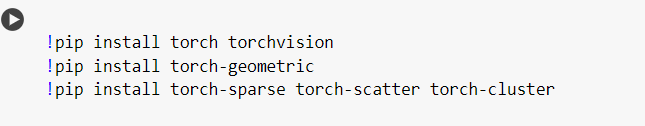
که خروجی آن به صورت زیر میشود :



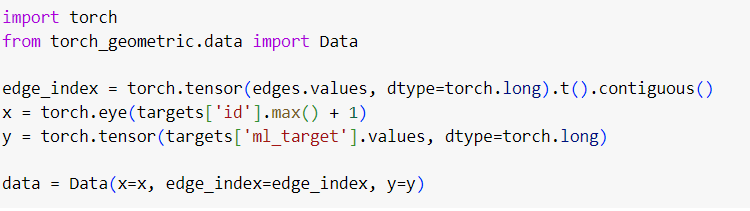
که به وضوح ناتراز بودن را نشان میدهد در ادامه سعی میکنیم این مورد را برطرف کنیم. برای این مورد از stratification استفاده میکنیم، بدین صورت است که از این برابر بودن در تست ترین و ولیدیشن مطمئن میشود.(البته کد آن کمی بعدتر ایمپلنت شده) ولی فقط همان خط ها را میاورم :



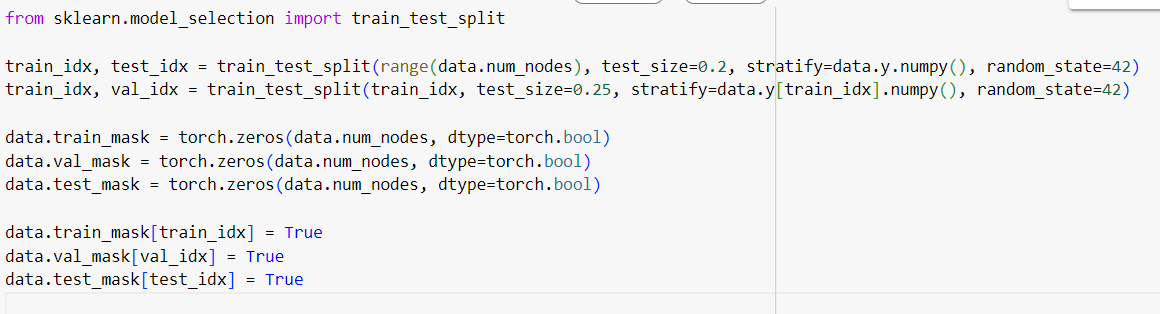
**ج)** خب قسمت اصلی کار در اینجاست، ابتدا برای کار با شبکه‌های عصبی به سراغ Pytorch میرویم و موارد لازم را در زمینه با GCNN نصب میکنیم :



در ادامه به استفاده از لایبرری geometric آن میپردازیم که بتوانیم استراکچر گراف را در آن تعریف کنیم. سپس دیتای یالها را به تنسور تبدیل میکنیم، و یک identity matrix میسازیم که هر سطر وان هات کودینگ یک نود خاص است(تا بتوانیم فیچرهای هر نود را در بیاریم) در نهایت یک Data میسازیم که تمامی این اطلاعات را دارد و به راحتی میتوان در GCN از آن استفاده کرد.



در ادامه عمل جداسازی دسته های مختلف را داریم و بعد از آن برای هر ست یک ماسک میسازیم که به صورت دیفالت همش False است. در ادامه همین ماسک ها را بر روی ایندکس ها میندازیم ( آن ایندکس ها True میشوند) .(انتخاب را در ادامه راحت میکند)



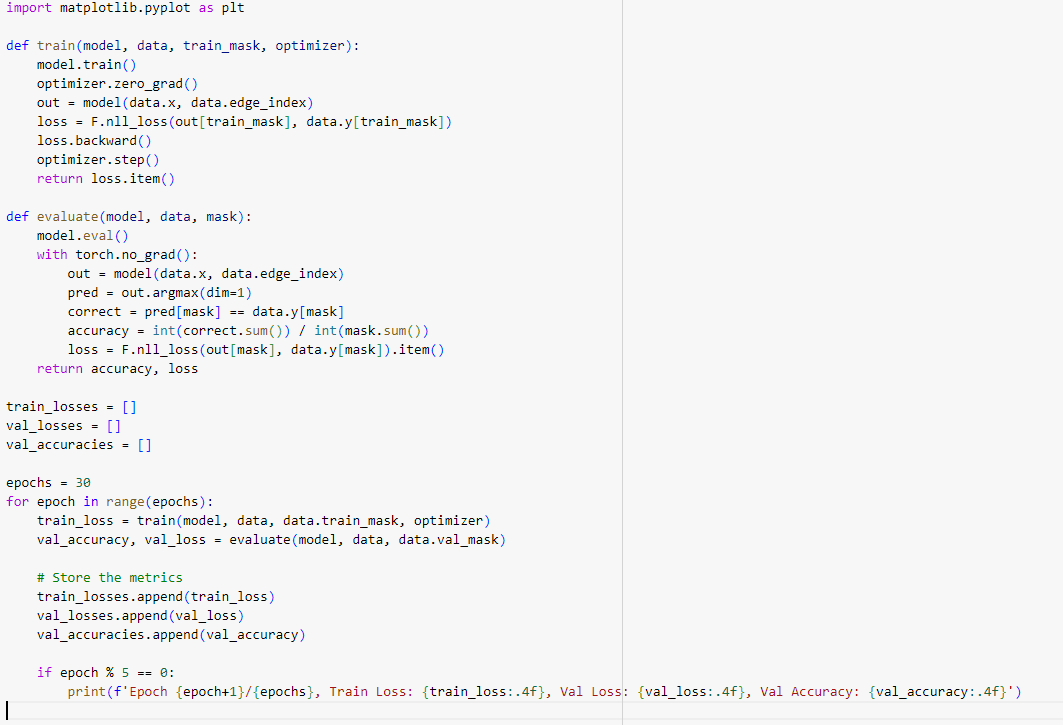
در ادامه به سراغ تعریف مدل میرویم که دو لایه ساده دارد که هر کدام از نوع GCNConv هستند لایه اول 16 خروجی و لایه دوم به تعداد کلاس که 2 است خروجی دارد(میتوانست پیشرفته تر باشد ولی همین هم جوابگو مسئله بود که در ادامه نتایج را میبینیم.)

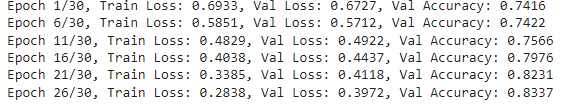
تابع forward هم تعریف میکنیم، که نحوه جریان دیتا در لایه ها را مشخص میکند(با نود و یال) سافتمکس در انتها مشخص شده است و بعد از لایه اول هم یک اکتیویشن فانکشن رلو داریم( و یک دراپ اوت بعد آن )



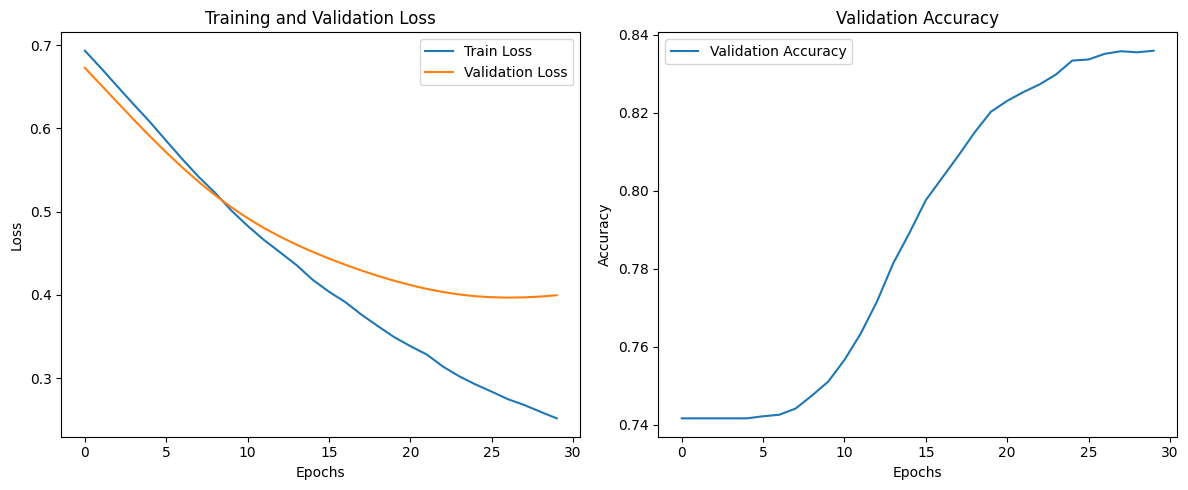
در گام آخر هم train وevaluation را داریم، که بدین صورت است که ابتدا مدل را در فاز آموزش قرار میدهیم، سپس گرادیان های قبلی را صفر میکنیم، خروجی مدل را بدست میاوریم و لاس را محاسبه میکنیم گرادیان را با loss.backward محاسبه میکنیم و پارامترهای مدل را آپدیت میکنیم.

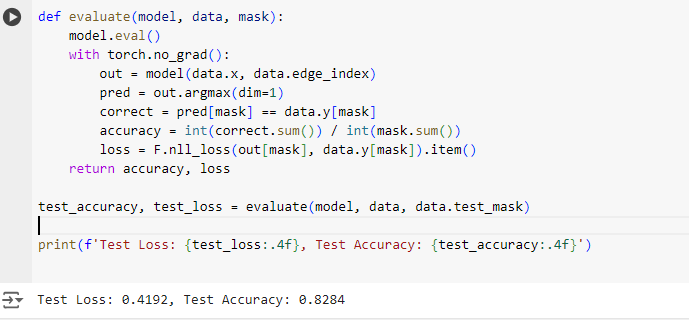
در فانکشن evaluate مدل را بر روی ماسک ها ارزیابی میکنیم ابتدا در مود آن قرار میگیرد، خروجی را پیدا میکند و کلاس پیشبینی را بدست میاورد و بر مبنای آن دقت و لاس را بدست میاورد.





**د)** در گام نهایی لازم بود تا یک سری نمودار جهت کارایی مدل نمایش دهیم همچنین بر روی ست تست خروجی بگیریم که به صورت زیر شد :





**بخش تشریحی)**

1) شبکه‌های عصبی گرافی در این زمینه به دلایل متعددی مفید هستند. یکی از دلایل اصلی این است که GNN‌ها قادرند روابط و تعاملات پیچیده بین نودها (کاربران و محصولات) را در یک گراف مدل‌سازی کنند، که این امر برای یک سامانه توصیه‌گر که باید ترجیحات کاربران و شباهت‌های محصولات را مدل کند، حیاتی است. علاوه بر این، GNN‌ها می‌توانند انواع مختلفی از داده‌ها مانند رفتار کاربران (مشاهده، خرید، امتیازدهی) و ویژگی‌های محصولات (از کاتالوگ) را ترکیب کنند و این رویکرد کلی‌نگر به مدل اجازه می‌دهد تا پیشنهادات دقیق‌تری ارائه دهد. از سوی دیگر، GNN‌ها می‌توانند بازنمایی‌های پنهان غنی از کاربران و محصولات یاد بگیرند که می‌تواند به پیش‌بینی انتخابات کاربران برای محصولات دیده نشده کمک کند. همچنین، GNN‌ها اطلاعات را در طول گراف منتشر می‌کنند، که به مدل اجازه می‌دهد تا نه تنها تعاملات مستقیم بلکه روابط غیرمستقیم را نیز در نظر بگیرد و کیفیت توصیه‌ها را بهبود بخشد.

2) برای ساخت گراف به عنوان ورودی برای GNN، ابتدا دو نوع نود تعریف می‌کنیم: کاربران و محصولات. سپس یال‌ها را بر اساس تعاملات ایجاد می‌کنیم؛ برای مثال، یک یال بین یک کاربر و یک محصول ایجاد می‌شود اگر کاربر محصول را مشاهده، خرید یا امتیازدهی کرده باشد. وزن یال می‌تواند نوع و تعداد تعامل را نشان دهد، مثلاً مشاهده ممکن است وزن کمتری نسبت به خرید داشته باشد. ویژگی‌های نودها نیز باید مشخص شوند؛ نودهای کاربر می‌توانند شامل اطلاعات شخصی کاربر، تاریخچه خریدهای گذشته و الگوهای مشارکت باشند، در حالی که نودهای محصول می‌توانند ویژگی‌هایی مانند دسته‌بندی، قیمت، توضیحات و سایر ویژگی‌ها را شامل شوند. علاوه بر این، به صورت اختیاری، یال‌ها می‌توانند ویژگی‌هایی مانند نوع تعامل و زمان‌بندی داشته باشند.

3) بازنمایی‌های به دست آمده از GNN می‌توانند به چندین روش در بهبود پیشنهادات کمک کنند. ابتدا، این شبکه embedding‌هایی برای نودهای کاربر ایجاد می‌کند که ترجیحات و الگوهای رفتاری آن‌ها را در بر می‌گیرد. به طور مشابه، GNN تعبیه‌هایی برای نودهای محصول تولید می‌کند که ویژگی‌های آن‌ها و روابطشان با سایر محصولات و کاربران را منعکس می‌کند. برای پیشنهاد محصولات، می‌توان شباهت بین embedding‌های کاربر و embedding محصول را محاسبه کرد و محصولاتی با بالاترین امتیاز شباهت به تعبیه کاربر پیشنهاد می‌شوند. در نهایت، با استفاده از این موارد پنهان، سیستم می‌تواند پیشنهادات بسیار شخصی‌سازی شده‌ای ارائه دهد که تعاملات مستقیم و غیرمستقیم را نیز در نظر می‌گیرد.

4) یکی از چالش‌های اصلی مقیاس‌پذیری است؛ GNN‌ها می‌توانند از نظر محاسباتی سنگین باشند، به خصوص با تعداد زیادی کاربر و محصول. برای حل این مشکل، می‌توان از تکنیک‌هایی مانند نمونه‌گیری (GraphSAGE)مثلاً برای مدیریت گراف‌های بزرگ به صورت کارآمدتر استفاده کرد و همچنین از چارچوب‌های محاسبات توزیعی بهره برد. چالش دیگر مشکل شروع سرد است؛ یعنی کاربران و محصولات جدید که هیچ تعاملی ندارند، داده‌های کمی دارند که این امر یادگیری تعبیه‌های معنادار را دشوار می‌کند. برای مقابله با این مشکل، می‌توان از مدل‌های ترکیبی که GNN‌ها را با فیلترهای مبتنی بر محتوا ترکیب می‌کنند، استفاده کرد.