a) هسته Contrastive Learning، Noise Contrastive Estimator Loss است که به صورت زیر است؟

$$NCE_{Loss} = -\log \frac{\exp(sim(g(x), g(x^+)))}{\exp(sim(g(x), g(x^+))) + \sum_{k=1}^{K} \exp(sim(g(x), g(x_k^-)))}$$

در SimCLR و موارد مشابه هم از تابع Loss هایی به این فرم استفاده می‌کنند و چون داده ها بدون برچسب هستند، برای تشکیل تابع هزینه نیاز به Negative Pair ها دارند. این فرم تابع هزینه ها باعث می‌شوند Positive Pair ها بازنمایی نزدیک همدیگر و Negative Pair ها را مجبور به فاصله و مستقل بودن از یکدیگر میکند.

اگر Negative Pair ها نبودند (یعنی عبارت مخرج در تابع Loss وجود نداشت) یک راه ساده این بود که SimCLR تمام بازنمایی ها را به یک نقطه ببرد که این صحیح نیست.

b) مشکل SimCLR در صورت استفاده نکردن از Negative Pair ها این است که در بهینه بد مثل یکسان شدن تمام بازنمایی ها ممکن است گیر کند. BYOL با رفع این مشکل که بازنمایی ها یکسان شود، از Negative pair ها استفاده نمی‌کند. برای این کار، از فیکس در نظر گرفتن شاخه target و عبور نکردن گرادیان از این شاخه استفاده می‌کند و گرادیان فقط از شاخه online عبور می‌کند تا بازنمایی view1 به بازنمایی view2 نزدیک تر شود. (assymetric است)

c) با توجه به اینکه BYOL از Negative examples استفاده نمی‌کند برای همین انتظار می‌رود نسبت به batch size کوچک مقاوم تر باشد اما به دلیل وجود batchnormalization در encoder برای بچ سایزهای کوچک (در paper، بچ سایز 128 نسبت به 256 بسیار افت داشته اما از 256 تا 4096 نسبتاً ثابت است) افت عملکرد دارد. هم چنین در paper ذکر شده است که BYOL سعی در حفظ اطلاعات شبکه target برای بهبود پیش بینی های شبکه‌ی آنلاین دارد بنابراین حتی اگر augmented view های یک تصویر با اطلاعات کمی مانند color histogram هم قابل تشخیص باشند، BYOL هم چنان می‌تواند اطلاعات اضافی برای بازنمایی را نگه دارد.

d) crop های با سایز کوچک از تصاویر ممکن است شامل اطلاعات متفاوتی از یکدیگر باشند برای همین در paper معرفی شده برای رفع این مشکل سعی بر افزایش فاصله بازنمایی های crop های با سایز کوچک (برای حفظ diversity بازنمایی های local objects local crops) و نزدیک کردن بازنمایی ها بین local و global crop ها و همچنین بازنمایی بین global crop ها دارد. (برای view invariance)

2

a)

1	2	3	4	5	6	7	8	مرکز
1	1	1	1	1	1	1	1	1
1	2	2	2	3	2	2	1	1
2	3	4	5	5	5	3	2	3
3	6	8	9	13	8	7	3	5

گراف سمت چپ

1	2	3	4	5	6	7	8	9	مرکز
1	1	1	1	1	1	1	1	1	1
1	2	2	2	3	3	2	2	2	1
2	3	4	5	6	7	5	5	4	3
3	6	8	10	15	16	11	11	10	6

گراف سمت راست.

۳ لایه برای متفاوت شدن بازنمایی گره مرکز نیاز است.

b)

$$M = D^{-1}A = \begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

ماتریس انتقال $\rightarrow$

$$M = \begin{array}{c|c} & \begin{matrix} 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

stationary Distribution :

$$\sum_{i \in V} \pi(i) = 1$$

$$\pi(i) = \frac{deg(i)}{\sum_{j \in V} deg(j)} = \frac{deg(i)}{2|E|}$$

$$\Rightarrow r = \begin{bmatrix} 1/4 & \frac{1}{4} & \frac{3}{8} & \frac{1}{8} \end{bmatrix}$$

c)

i)

$$h_i^{l+1} = \frac{1}{|N_i|}\sum_{j\in N_i} h_j^l \rightarrow h_i^{l+1} = D^{-1}A$$

ii)

$$h_i^{l+1} = \frac{1}{2}h_i^l + \frac{1}{2|N_i|}\sum_{j\in N_i} h_j^l \rightarrow h_i^{l+1} = \frac{1}{2}(I + D^{-1}A)$$

d) در این سوال از Notation مقاله Deeper insight Graph convolutional Networks for semi-supervised Learning Qimai Li, ... استفاده شده است.

$$h^{l+1} = D^{-1}Ah^l$$

$$= D^{-1}(D + A - D)h^l$$

$$\xrightarrow{L=A-D} = (I + D^{-1}L)h^l$$

$$L_{rw} = D^{-1}L$$

$$\Rightarrow h^{l+1} = \lim_{l\to\infty}(I - L_{rw})^l h^0$$

در مقاله ثابت می‌شود که اگر گراف شامل اجزای دوبخشی نباشد، ترکیب گراف چندگانه به یک ترکیب خطی همگرا می‌شود ($\{1^{(i)}\theta_i\}_{i=1}^K$) :

$$1_j^{(i)} = \begin{cases} 1, & v_j \in c_i \\ 0, & v_j \notin c_i \end{cases} \rightarrow$$ مشخص می‌کند که نود j در مؤلفه‌ی c_i وجود دارد یا خیر و $\theta \in \mathbb{R}^n$

$$\Rightarrow \lim_{l\to\infty}(I - L_{rw})^l h^0 = [1^{(1)}, 1^{(2)}, \dots, 1^{(K)}]\theta$$

اثبات: فرض کنید v_k، بردار ویژه L_{rw} باشد، آنگاه $(I - L_{rw})v_k = v_k - \lambda_k v_k = (I - \lambda_k)v_k$. یعنی مقادیر ویژه Λ' ماتریس $(I - L_{rw})$ برابر $I - \Lambda_{rw}$ است.

می‌دانیم مقادیر ویژه لاپلاسین گراف نرمال شده و لاپلاسین گراف نرمال‌شده متقارن برابرند. مقادیر ویژه NGL بین $[0, 2)$ مثل SNGL هستند.

پس مقادیر ویژه $(I - L_{rw})$ بین $(-1, 1]$ هستند و λ'_n بزرگترین مقدار ویژه مربوط به مقدار ویژه 0 در L_{rw} است و بردار ویژه‌ی متناظر با آن $1^{(i)}$ است.

$$\lim_{l\to\infty}(I - L_{rw})^l h^0 = \lim U\Lambda^l U^{-1} = [1^{(1)}, 1^{(2)}, \dots, 1^{(K)}]\theta$$

e) در الگوریتم BFS به علت اصطلاح 1-hop Neighborhood نیاز داریم بنابراین GCN تک لایه کافی است.

Aggregation method:

$$m_u^l = AGGREGATE(\{h_u^l, \forall v \in N(u)\}) = \max\{h_v^l \mid v \in N(u)\}$$

update:

$$h^{l+1} = \max\{h_v^{(l)}, m_u^l\}$$

→ گره‌ای قبلاً بازدید شده و embedding آن ۱ شده است.
یا یکی از همسایه‌هایش بازدید شده و در مرحله فعلی اکتشاف می‌شود.