A Experimental Settings

A.1 Counterparts and HyReaL Settings

In Comparison approaches, we follow their original settings. For traditional methods, the K-Means clusters the features without graph structure, and the Spectral clustering clusters the graph structure without feature matrix. They are executed 10 times for average scores. For conventional methods, we perform 200 epochs of unsupervised training of the GAE and VGAE, then use K-Means to cluster the generated embedding. For advanced and state-of-the-art clustering approaches, we reproduce their source code by following the original parameter settings in the source codes.

There are some hyper-parameters and settings of our method, i.e., the layer number, pre-training learning rate, pre-training iteration number, learning rate, iteration number, model regularization trade-off α, and representation embedding loss trade-off β. We set Adam optimizer during experiments. The activation function of the graph encoder is ReLU for each layer. $\mathcal{L}_{reg}$ in the loss is the regularization of model, and L1 regularization is utilized. In the pre-training process, the hyper-parameter β is set to 0.0001. For ten datasets, the neuron number of layers, the pre-training learning rate, pre-training iteration number, and iteration number are set to $[512, 256, 128]$, 10^{-4}, 10, and 4, respectively. The learning rate is set to 0.00001 for CITESEER, 0.0002 for ACM, WIKI, and AMAP, 0.0004 for DBLP, FILM, CORNELL, CORA, WISC, and, UAT. The α is set to 10^{-4} for CORNELL, 2×10^{-4} and UAT, 5×10^{-4} for CITESEER, FILM, and WISC, 10^{-5} for CORA and DBLP, 10^{-6} for ACM, WIKI, AMAP. The β is set to 2^{-10} for WIKI, DBLP, CORNELL, CORA, WISC, 2^{-12} for ACM, CITESEER, 2^{-30} for UAT, 0 for AMAP.

A.2 Description of Validity Metrics

We provide a more detailed description of validity metrics, which are Accuracy (ACC), Normalized Mutual Information (NMI), and Adjusted Rand Index (ARI) [Zhou *et al.*2022].

ACC is a straightforward measure that calculates the percentage of correctly classified data points in the clustering results compared to ground truth. A higher accuracy indicates better performance. Given ground truth labels $Y = \{y_i | 1 \leq i \leq n\}$ and the predicted clusters $\hat{Y} = \{\hat{y_i} | 1 \leq i \leq n\}$, ACC is computed as

$$ACC(\hat{Y}, Y) = \max \frac{1}{n} \sum_{i=1}^{n} 1\{y_i = \hat{y_i}\}. \tag{1}$$

NMI quantifies the amount of shared information between two clusters. It ranges from 0 to 1, where 1 indicates perfect agreement and vice versa. Higher NMI values indicate better clustering performance. The NMI can be computed by

$$NMI(\tilde{Y}, Y) = \frac{T(\tilde{Y}; Y)}{\frac{1}{2}\left[H(\tilde{Y}) + H(Y)\right]}, \tag{2}$$

Table A.1. The Wilcoxon signed rank test with 95% confidence interval. The symbols "+" and "−" indicate the rejection and acceptance of the null hypothesis.

Method	ACC	NMI	ARI
K-Means	+	+	+
GAE	+	+	+
ARVGAE	+	+	+
DAEGC	+	+	+
CCGC	−	+	+
DFCN	+	+	+
EGAE	+	+	+
CONVERT	−	+	+
SCDGN	−	+	+
MAGI	+	−	−
GLAC	+	+	+

where $H(Y)$ is entropy of Y and $T(\tilde{Y};Y)$ is mutual information between $\tilde{Y}$ and Y.

ARI measures the similarity between two clusters, taking into account both true positive and true negative matches while correcting for chance. It produces a value between -1 and 1. An ARI value close to 1 suggests strong agreement, close to 0 indicates random agreement, and negative values indicate disagreement. A higher ARI value indicates better clustering performance, and the ARI can be computed as

$$ARI = \frac{RI - \mathbb{E}(RI)}{\max(RI - \mathbb{E}(RI))}, \tag{3}$$

where

$$RI = \frac{TP + TN}{C_n^2}. \tag{4}$$

Here, TP and FP respectively denote the number of true positive pairs and true negative pairs, and C_n^2 is the number of possible object pairs.

A.3 Settings of the Wilcoxon Signed-ranks Test

Here, we provide experimental settings of the Wilcoxon signed-ranks test for the results in Table A.1 of the submitted paper.

The Wilcoxon signed-ranks test is a non-parametric alternative to the paired t-test. It ranks the differences in performances of two classifiers for each dataset, ignoring the signs, and compares the ranks for the positive and the negative differences [Demsar2006]. In general, the Wilcoxon signed-ranks test is used when we have paired data and try to observe if there is a significant change. If the test statistic is smaller than the critical value from a table (or if the p-value is below a chosen significance level), we can reject the null hypothesis, which suggests a significant difference between the paired data.

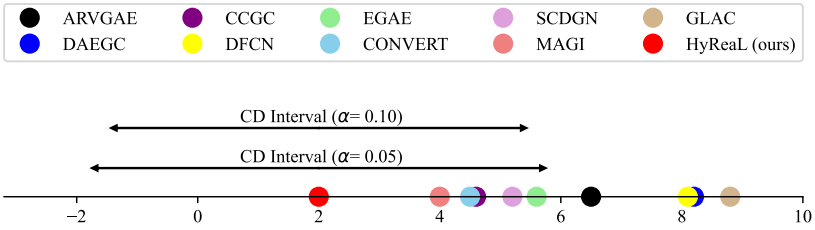

Fig. A.1. Visualization of Bonferroni-Dunn (BD) test at confidence intervals 90% and 95%.

The procedures of the Wilcoxon signed-ranks test are: 1) Calculate the differences between paired observations. 2) Rank these differences in absolute rank values. 3) Assign positive or negative signs to the ranks based on the direction of the differences. 4) Sum the ranks of positive and negative differences separately. The smaller of the two sums is utilized for the test. If the smaller value is smaller than the critical value, we will reject the null hypothesis.

In our experiment, the Wilcoxon signed-ranks test is conducted to compare our method with other methods under different validity metrics on all the ten datasets. The procedures are as follows: 1) Formulate the hypothesis where the null hypothesis is that HyReaL does not exhibit a significant difference, or perform equally, compared to other models under a specific validity metric. The alternative hypothesis is that HyReaL significantly outperforms other models. 2) Set the significance level at 0.01. 3) Calculate the p-value of the compared model performance. 4) Obtain the test results. If the p-value is less than the chosen significance level, we reject the null hypothesis, and vice versa, where a rejection suggests that HyReaL significantly outperforms the compared model.

B Algorithm and Complexity Analysis of the HyReaL

B.1 Algorithm of HyReaL

The algorithm process of HyReaL is shown in Algorithm 1.

B.2 Rationality of Loss Function

Remark 1. **Rationality of the loss function.** It is noteworthy that $\mathbf{L}$ obtained based on $\hat{\mathbf{A}}$ is the key factor to influence the accuracy of clustering. To learn more powerful $\hat{\mathbf{A}}$, the model is designed with a higher DoF in feature encoding facilitated based on the quaternion product. On such basis, the training process comprehensively takes into account the attribute information by the FVP and QGE, preserves the graph topology by the graph reconstruction decoder, and customizes the general clustering-friendly representation by introducing the clustering-oriented loss $\mathcal{L}_{sc}$.

Algorithm 1 HyReaL: Hyper-complex space Representation Learning.

Input: Attributed graph $G = \{\mathbf{A}, \mathbf{X}\}$; Cluster number k; Loss weights α and β.
Output: k non-overlapping sub-graphs $\{G_1, G_2, ..., G_k\}$.

Convert the adjacency matrix $\mathbf{A}$ into symmetric normalized Laplacian matrix $\tilde{\mathbf{A}}$;
repeat
  Project $\mathbf{X}$ into four views $\mathbf{F}_\triangleright$ by Eq. (1) and form a feature quaternion $\mathbf{F}$ as shown in Eq. (2);
  Encode $\mathbf{F}$ using quaternion graph encoders defined by Eqs. (3) and (4);
  Obtain the output embeddings $\mathbf{\Gamma}$ by the quaternion fusion operator defined in Eq. (5);
  Reconstruct the adjacency matrix $\hat{\mathbf{A}}$ from $\mathbf{\Gamma}$ according to Eq. (6);
  Compute the value of objective function $\mathcal{L}$ according to Eqs. (7) - (10);
  Update learnable parameters $\mathbf{W}_\triangleright^L, \mathbf{B}_\triangleright^L$ and $\mathbf{W}_l^Q$.
until maximum iterations reached
Perform spectral clustering to solve Eq. (10) based on $\hat{\mathbf{A}}$ reconstructed from the final $\mathbf{\Gamma}$.

C Complementary Experimental Results

C.1 Complementary Separability and Universality Evaluation of Representations

The complementary experiments of separability and universality evaluation of representations, the plot is shown in Figure A.6.

C.2 Complementary Effectiveness of QGE on Mitigating Over-smoothing

The complementary plots of effectiveness of QGE on mitigating over-smoothing, the plot is shown in Figure A.7.

C.3 Supplemental Ablation Studies

Three observations are provided below: 1) HyReaL performs better than the Baseline in 29 out of 30 comparisons, clearly illustrating the adaptability of FVP and QGE modules in the learning. 2) HyReaL performs better than HyReaL w/o FVP in 27 out of 30 comparisons. This evidently indicates that FVP is a necessary pre-phase of QGE. HyReaL w/o FVP makes the four MLPs unlearnable, thus FVP degrades to a random projection of the input attributes, which surely loses the ability to provide suitable feature quaternions for QGE. 3) HyReaL outperforms HyReaL w/o QGE in 22 out of 30 comparisons. This indicates that QGE is effective in aggregating the node information and preventing the OD effect. Without the quaternion transformation in QGE, HyReaL cannot facilitate a high DoF representation learning of attributes to offset the OD effect. As a result, the embeddings of two very dissimilar but graph-adjacent nodes may be homogeneous, which hinders accurate clustering. 4) HyReaL performs better

than HyReaL w/o β in 23 out of 30 comparisons. This clearly claims that the cluster-oriented loss contributes to enhancing the representation ability. That is, the objective function provides more clustering-friendly optimization guidance in optimization.

C.4 The Results of Wilcoxon Signed Rank Test on Comparative Experiments

Table A.1 is the Wilcoxon signed rank test of comparative experiments results.

C.5 Bonferroni-Dunn Test of Comparison Experiment

In order to comprehensively demonstrate the superiority of our model compared to other methods, we conduct the Bonferroni-Dunn Test (BD test) [Demsar2006] based on the average rank (i.e., the 'AR' row) of the comparative experimental results in Table 1 of the main paper.

The Bonferroni-Dunn test is used to compare an algorithm with the remaining $k-1$ counterparts. It involves comparing the differences in average ranks of various methods with a certain threshold value called Critical Difference (CD). The CD is defined as:

$$CD = q_\lambda \sqrt{\frac{p(p+1)}{6N}}, \tag{5}$$

where q_λ is critical values for the BD test, p is the number of compared methods, and N is the number of dataset. If the rank difference between the two methods is higher than the CD, it indicates that the method with the higher average rank is statistically superior to the one with the lower average rank. Conversely, if the difference is lower than the CD, it suggests that there is no significant performance difference between the two methods.

Our BD test conduction procedures are as follows. 1) We obtain the ranks of the methods under all three validity metrics on all ten datasets. 2) The ranks under the three metrics are averaged to an overall rank of the corresponding method w.r.t. each certain dataset. 3) The average ranks on ten datasets are further averaged to an overall average rank of the methods, which are shown in Table 1 of the main paper.

According to [Demsar2006], we set the confidence intervals to 90% and 95%, and compute the CD by

$$CD_{0.10} = 3.4378, \tag{6}$$

and

$$CD_{0.05} = 3.7546, \tag{7}$$

where the $q_{0.10}$ and $q_{0.05}$ of ten classifiers are 2.539 and 2.773 according to Table 5(b) in reference [Demsar2006], the number of datasets N is 10, and the number of compared methods p is 10. Overall, it can be observed that HyReaL performs significantly better than the seven methods, as shown in Figure A.1.

C.6 Training Convergence Evaluation

To demonstrate the convergence of our model, we show its convergence curves on all the ten benchmark datasets in Figure A.2.

The overall trend of the loss convergence curves indicates a steady decrease in loss, which suggests that the model can effectively learn from the training data. Although there are minor fluctuations in the loss curves on some datasets, the loss decreasing tends stable when approaching the pre-set 50 epoch of training. In summary, the training convergence evaluation illustrates that our model can be effectively trained for learning representation and clustering.

C.7 Sensitivity Evaluation of Hyper-Parameters

The sensitivity of HyReaL to the trade-off hyper-parameters α and β is evaluated on the datasets as shown in Figure A.3. Note that when evaluating sensitivity to one parameter, another one is fixed at the corresponding settings in Appendix A.1. From the results, it is not surprising that a too-large value of α or β leads to generating objective biased representations such that HyReaL obtains undesired clustering performance. The results also confirmed that HyReaL is insensitive to α and β in the value range around the parameter settings adopted for the aforementioned experiments.

C.8 Visual Results

The supplementary t-SNE visualization results of the representations learned by different methods on the ACM and DBLP datasets are shown in Fig A.4.

To intuitively compare the ablated versions of HyReaL, the representations learned by them and HyReaL are also compared using t-SNE on the CORA dataset in Figure A.5.

For all the visualization results in this section, the observations and conclusions are consistent with the corresponding results in the main paper, so we do not provide redundant discussions here.

D Discussion about Remark and Proof

D.1 Detailed Remark of Learning Ability

We provide a more detailed analysis of "Remark 1" in Section 3.2 of the submitted paper. The more detailed Remark 1 is given below.

Remark 2. **Degree of Freedom.** According to Eq. (3) in main paper, learnable parameters in our model, i.e., $\mathbf{W}_{\mathbb{H}}^{Q} = \{\mathbf{W}_{r}^{Q}, \mathbf{W}_{x}^{Q}, \mathbf{W}_{y}^{Q}, \mathbf{W}_{z}^{Q}\}$, yields 16 pairs of feature interaction. In contrast, realizing the same scale interaction in real-value space requires 4 times of parameters. This illustrates the learning efficiency of the proposed model. Detailed analysis is given below.

Given model input

$$\mathbf{F} = \{\mathbf{F}_r, \mathbf{F}_x, \mathbf{F}_y, \mathbf{F}_z\}, \tag{8}$$

where $\mathbf{F} \in \mathbb{H}^{n\times(4\times\hat{d})}$, $\hat{d}$ indicates the dimension of input. Then, we define the learnable parameters of quaternion representation as $\mathbf{W}_{\mathbb{H}}^{Q} \in \mathbb{H}^{(4\times\hat{d})\times(4\times\tilde{d})}$, which contains four part of parameters $\{\mathbf{W}_r^Q, \mathbf{W}_x^Q, \mathbf{W}_y^Q, \mathbf{W}_z^Q\}$, where $\tilde{d}$ is the output dimension, and $\mathbf{W}_i^Q \in \mathbb{H}^{\hat{d}\times\tilde{d}}$ with $i \in \{r, x, y, z\}$.

According to the Hamilton product in the quaternion system, the learnable parameters let the features in $\mathbf{F}$ interact by

$$\begin{aligned}
\mathbf{F}^Q &= \mathbf{F} \otimes \mathbf{W}^Q \\
&= \mathbf{W}_r^Q\mathbf{F}_r - \mathbf{W}_x^Q\mathbf{F}_x - \mathbf{W}_y^Q\mathbf{F}_y - \mathbf{W}_z^Q\mathbf{F}_z \\
&+ \mathbf{W}_x^Q\mathbf{F}_r + \mathbf{W}_r^Q\mathbf{F}_x - \mathbf{W}_z^Q\mathbf{F}_y + \mathbf{W}_y^Q\mathbf{F}_z\,, \\
&+ \mathbf{W}_y^Q\mathbf{F}_r + \mathbf{W}_z^Q\mathbf{F}_x + \mathbf{W}_r^Q\mathbf{F}_y - \mathbf{W}_x^Q\mathbf{F}_z \\
&+ \mathbf{W}_z^Q\mathbf{F}_r - \mathbf{W}_y^Q\mathbf{F}_x + \mathbf{W}_x^Q\mathbf{F}_y + \mathbf{W}_r^Q\mathbf{F}_z
\end{aligned} \tag{9}$$

where $\mathbf{F}^Q \in \mathbb{H}^{n\times(4\times\tilde{d})}$. It is intuitive that such an operation yields learning with a 16-Degree of Freedom (DoF).

In the following, we design a real-value model with the same DoF, and observe how many parameters are required for comparison. For intuitive comparison, we define the parameters of the real-value model in a similar form as that of the quaternion model, i.e., $\mathbf{W}_i^R \in \mathbb{R}^{4\times(\hat{d}\times\tilde{d})}$ with $i \in \{r, x, y, z\}$. The superscript R indicates that these are the parameters of the real-value model. Accordingly, all the features in $\mathbf{F}$ interact through the parameters by

$$\mathbf{F}^R = [\mathbf{F}_r\mathbf{W}_r^R, \mathbf{F}_x\mathbf{W}_x^R, \mathbf{F}_y\mathbf{W}_y^R, \mathbf{F}_z\mathbf{W}_z^R], \tag{10}$$

where $\mathbf{F}^R \in \mathbb{R}^{n\times(4\times\tilde{d})}$ is the output matrix, and $\mathbf{W}_r^R, \mathbf{W}_x^R, \mathbf{W}_y^R, \mathbf{W}_z^R$ can be written as

$$\begin{aligned}
\mathbf{W}_r^R &= [\mathbf{W}_1^R, \mathbf{W}_2^R, \mathbf{W}_3^R, \mathbf{W}_4^R] \\
\mathbf{W}_x^R &= [\mathbf{W}_5^R, \mathbf{W}_6^R, \mathbf{W}_7^R, \mathbf{W}_8^R] \\
\mathbf{W}_y^R &= [\mathbf{W}_9^R, \mathbf{W}_{10}^R, \mathbf{W}_{11}^R, \mathbf{W}_{12}^R] \\
\mathbf{W}_z^R &= [\mathbf{W}_{13}^R, \mathbf{W}_{14}^R, \mathbf{W}_{15}^R, \mathbf{W}_{16}^R].
\end{aligned} \tag{11}$$

Accordingly, the output feature $\mathbf{F}^R$ is written as

$$\mathbf{F}^R = \begin{bmatrix}
\mathbf{W}_1^R\mathbf{F}_r + \mathbf{W}_2^R\mathbf{F}_r + \mathbf{W}_3^R\mathbf{F}_r + \mathbf{W}_4^R\mathbf{F}_r \\
\mathbf{W}_5^R\mathbf{F}_x + \mathbf{W}_6^R\mathbf{F}_x + \mathbf{W}_7^R\mathbf{F}_x + \mathbf{W}_8^R\mathbf{F}_x \\
\mathbf{W}_9^R\mathbf{F}_y + \mathbf{W}_{10}^R\mathbf{F}_y + \mathbf{W}_{11}^R\mathbf{F}_y + \mathbf{W}_{12}^R\mathbf{F}_y \\
\mathbf{W}_{13}^R\mathbf{F}_z + \mathbf{W}_{14}^R\mathbf{F}_z + \mathbf{W}_{15}^R\mathbf{F}_z + \mathbf{W}_{16}^R\mathbf{F}_z
\end{bmatrix}^\top. \tag{12}$$

Obviously, there are 16 parameter matrices that are of the same size of $\mathbf{W}_i^R \in \mathbb{R}^{4\times(\hat{d}\times\tilde{d})}$, and it can be concluded that if a real-value model is adopted to realize the same DoF as that of the quaternion-value model, a four-time model

parameters scale will be involved. In other words, with the same number of parameters, the quaternion-value model can achieve a higher DoF than real-value models for more informative representation and cluster learning.

D.2 Proof of Degree of Freedom

According to above discussions, the learnable parameters $\mathbf{W}^Q$ can generate 16 features learning pairs. In the same DoF representation, the number of real-value parameters is four times of the quaternion representation. Thus, with the same number of parameters, the quaternion representation model can achieve a four times DoF comparing to the conventional real-value representation learning, and thus helps to more thoroughly explore features informative and benefit cluster learning.

Rigorous proof of the '4×DoF' is provided below. We follow the parameter initialization method in [Parcollet *et al.*2019] and initial our quaternion graph encoders. In order to simplify the mathematical expression and analyses, especially when dealing with operations involving complex numbers or quaternions, we introduce the polar coordinate form to represent the weight of quaternion. A generated quaternion weight w from a weight matrix $\mathbf{W}^Q$ has a polar form defined as:

$$w = |w|e^{q^{\triangleleft}_{img}\theta} = |w|(\cos(\theta) + q^{\triangleleft}_{img}\sin(\theta)), \tag{13}$$

with $q^{\triangleleft}_{img} = 0 + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$ a purely imaginary and normalized quaternion, and the angle θ is randomly generated in the interval $[-\pi, \pi]$. The imaginary components $x\mathbf{i}$, $y\mathbf{j}$, and $z\mathbf{k}$ are sampled from an uniform distribution in $[0, 1]$ to obtian q_{imag}. The parameter φ is a random number generated with respect to well-known initialization criterions (such as Glorot [Glorot and Bengio2010] or He algorithms [He *et al.*2015]). Therefore, w can be computed following:

$$\begin{aligned} w_{\mathbf{r}} &= \varphi\cos(\theta), \\ w_{\mathbf{i}} &= \varphi q^{\triangleleft}_{img\mathbf{i}}\sin(\theta), \\ w_{\mathbf{j}} &= \varphi q^{\triangleleft}_{img\mathbf{j}}\sin(\theta), \\ w_{\mathbf{k}} &= \varphi q^{\triangleleft}_{img\mathbf{k}}\sin(\theta). \end{aligned} \tag{14}$$

However, φ represents a randomly generated variable with respect to the variance of the quaternion weight and the selected initialization criterion. The initialization process follows [Glorot and Bengio2010] or [He *et al.*2015] to derive the variance of the quaternion-valued weight parameters. Indeed, the variance of $\mathbf{W^Q}$ has to be investigated:

$$\mathrm{Var}(\mathbf{W^Q}) = \mathbb{E}\left(|\mathbf{W^Q}|^2\right) - [\mathbb{E}(|\mathbf{W^Q}|)]^2, \tag{15}$$

$[\mathbb{E}(|\mathbf{W^Q}|)]^2$ **equals to 0 since the weight distribution is symmetric around 0.** Nonetheless, the value of $\mathrm{Var}(\mathbf{W^Q}) = \mathbb{E}\left(|\mathbf{W^Q}|^2\right)$ is not trivial in the case of quaternion-valued matrices. Indeed, $\mathbf{W^Q}$ follows a Chi-distribution with four

degrees of freedom (DoFs). Chi-distribution is often used to describe the distribution mode of the modulus length or Euclidean distance of multi-dimensional vectors. Thus, $\mathbb{E}\left(|\mathbf{W}^{\mathbf{Q}}|^2\right)$ is expressed and computed as follows:

$$\mathbb{E}\left(|\mathbf{W}^{\mathbf{Q}}|^2\right) = \int_0^\infty x^2 f(x)\mathrm{d}x, \tag{16}$$

with $f(x)$ is the probability density function with four DoFs. A four-dimensional vector $X = \{A, B, C, D\}$ is considered to evaluate the density function $f(x)$. X has components that are normally distributed, centered at zero, and independent. Then, A, B, C and D have density functions:

$$f_A(x;\sigma) = f_B(x;\sigma) = f_C(x;\sigma) = f_D(x;\sigma) = \frac{e^{-x^2/2\sigma^2}}{\sqrt{2\pi\sigma^2}}. \tag{17}$$

The 4D vector X has a length L defined as $L = \sqrt{A^2 + B^2 + C^2 + D^2}$ with a cumulative distribution function $F_L(x;\sigma)$ in the 4-sphere (n-sphere with $n = 4$) S_x:

$$F_L(x;\sigma) = \iiiint_{S_x} f_A(x;\sigma) f_B(x;\sigma) f_C(x;\sigma) f_D(x;\sigma)\mathrm{d}S_x, \tag{18}$$

where $S_x = \left\{(a, b, c, d) : \sqrt{a^2 + b^2 + c^2 + d^2} < x\right\}$ and $\mathrm{d}S_x = \mathrm{d}a\mathrm{d}b\mathrm{d}c\mathrm{d}d$. The polar representations of the coordinates of X in a 4-dimensional space are defined to compute $\mathrm{d}S_x$:

$$\begin{aligned} a &= \rho\cos\theta \\ b &= \rho\sin\theta\cos\phi \\ c &= \rho\sin\theta\sin\phi\cos\psi \\ d &= \rho\sin\theta\sin\phi\sin\psi, \end{aligned} \tag{19}$$

where ρ is the magnitude ($\rho = \sqrt{a^2 + b^2 + c^2 + d^2}$) and θ, ϕ, and ψ are the phases with $0 \le \theta \le \pi$, $0 \le \phi \le \pi$ and $0 \le \psi \le 2\pi$. Then, $\mathrm{d}S_x$ is evaluated with the Jacobian J_f of f defined as:

$$\begin{aligned} J_f = \frac{\partial(a, b, c, d)}{\partial(\rho, \theta, \phi, \psi)} = \frac{\mathrm{d}a\mathrm{d}b\mathrm{d}c\mathrm{d}d}{\mathrm{d}\rho\mathrm{d}\theta\mathrm{d}\phi\mathrm{d}\psi} = \begin{vmatrix} \frac{\mathrm{d}a}{\mathrm{d}\rho} & \frac{\mathrm{d}a}{\mathrm{d}\theta} & \frac{\mathrm{d}a}{\mathrm{d}\phi} & \frac{\mathrm{d}a}{\mathrm{d}\psi} \\ \frac{\mathrm{d}b}{\mathrm{d}\rho} & \frac{\mathrm{d}b}{\mathrm{d}\theta} & \frac{\mathrm{d}b}{\mathrm{d}\phi} & \frac{\mathrm{d}b}{\mathrm{d}\psi} \\ \frac{\mathrm{d}c}{\mathrm{d}\rho} & \frac{\mathrm{d}c}{\mathrm{d}\theta} & \frac{\mathrm{d}c}{\mathrm{d}\phi} & \frac{\mathrm{d}c}{\mathrm{d}\psi} \\ \frac{\mathrm{d}d}{\mathrm{d}\rho} & \frac{\mathrm{d}d}{\mathrm{d}\theta} & \frac{\mathrm{d}d}{\mathrm{d}\phi} & \frac{\mathrm{d}d}{\mathrm{d}\psi} \end{vmatrix} \\ = \begin{vmatrix} \cos\theta & -\rho\sin\theta & 0 & 0 \\ \sin\theta\cos\phi & \rho\sin\theta\cos\phi & -\rho\sin\theta\sin\phi & 0 \\ \sin\theta\sin\phi\cos\psi & \rho\cos\theta\sin\phi\cos\psi & \rho\sin\theta\cos\phi\cos\psi & -\rho\sin\theta\sin\phi\sin\psi \\ \sin\theta\sin\phi\sin\psi & \rho\cos\theta\sin\phi\sin\psi & \rho\sin\theta\cos\phi\sin\psi & \rho\sin\theta\sin\phi\cos\psi \end{vmatrix}. \end{aligned} \tag{20}$$

And,

$$J_f = \rho^3\sin^2\theta\sin\phi. \tag{21}$$

Therefore, by the Jacobian J_f, we have the polar form:

$$\mathrm{d}a\mathrm{d}b\mathrm{d}c\mathrm{d}d = \rho^3\sin^2\theta\sin\phi\mathrm{d}\rho\mathrm{d}\theta\mathrm{d}\phi\mathrm{d}\psi. \tag{22}$$

Then, writing Eq. (18) in polar coordinates, we obtain:

$$\begin{aligned}
F_L(x,\sigma) &= \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^4 \iiint\int_0^x e^{-a^2/2\sigma^2} e^{-b^2/2\sigma^2} e^{-c^2/2\sigma^2} e^{-d^2/2\sigma^2} \mathrm{d}S_x \\
&= \frac{1}{4\pi^2\sigma^4} \int_0^{2\pi}\int_0^{\pi}\int_0^{\pi}\int_0^{x} e^{-\rho^2/2\sigma^2} \rho^3 \sin^2\theta \sin\phi \mathrm{d}\rho\mathrm{d}\theta\mathrm{d}\phi\mathrm{d}\psi \\
&= \frac{1}{4\pi^2\sigma^4} \int_0^{2\pi} \mathrm{d}\psi \int_0^{\pi} \sin\phi \mathrm{d}\phi \int_0^{\pi} \sin^2\theta \mathrm{d}\theta \int_0^{x} \rho^3 e^{-\rho^2/2\sigma^2} \mathrm{d}\rho \\
&= \frac{1}{4\pi^2\sigma^4} 2\pi 2 \left[\frac{\theta}{2} - \frac{\sin 2\theta}{4}\right]_0^{\pi} \int_0^{x} \rho^3 e^{-\rho^2/2\sigma^2} \mathrm{d}\rho \\
&= \frac{1}{4\pi^2\sigma^4} 4\pi \frac{\pi}{2} \int_0^{x} \rho^3 e^{-\rho^2/2\sigma^2} \mathrm{d}\rho
\end{aligned} \tag{23}$$

Then,

$$F_L(x,\sigma) = \frac{1}{2\sigma^4} \int_0^{x} \rho^3 e^{-\rho^2/2\sigma^2} \mathrm{d}\rho. \tag{24}$$

The probability density function for X is the derivative of its cumulative distribution function, which by the fundamental theorem of calculus is:

$$\begin{aligned}
f_L(x,\sigma) &= \frac{\mathrm{d}}{\mathrm{d}x} F_L(x,\sigma) \\
&= \frac{1}{2\sigma^4} x^3 e^{-x^2/2\sigma^2}
\end{aligned} \tag{25}$$

The expectation of the squared magnitude becomes:

$$\begin{aligned}
\mathbb{E}\left(|\mathbf{W}^Q|^2\right) &= \int_0^{\infty} x^2 f(x) \mathrm{d}x \\
&= \int_0^{\infty} x^2 \frac{1}{2\sigma^4} x^3 e^{-x^2/2\sigma^2} \mathrm{d}x \\
&= \frac{1}{2\sigma^4} \int_0^{\infty} x^5 e^{-x^2/2\sigma^2} \mathrm{d}x
\end{aligned} \tag{26}$$

With integration by parts we obtain:

$$\begin{aligned}
\mathbb{E}\left(|\mathbf{W}^Q|^2\right) &= \frac{1}{2\sigma^4} \left(- x^4\sigma^2 e^{-x^2/2\sigma^2}\Big|_0^{\infty} + \int_0^{\infty} \sigma^2 4x^3 e^{-x^2/2\sigma^2} \mathrm{d}x \right) \\
&= \frac{1}{2\sigma^2} \left(- x^4 e^{-x^2/2\sigma^2}\Big|_0^{\infty} + \int_0^{\infty} 4x^3 e^{-x^2/2\sigma^2} \mathrm{d}x \right)
\end{aligned} \tag{27}$$

The expectation $\mathbb{E}\left(|\mathbf{W}^Q|^2\right)$ is the sum of two terms. The first one:

$$\begin{aligned}
- x^4 e^{-x^2/2\sigma^2}\Big|_0^{\infty} &= \lim_{x\to+\infty} -x^4 e^{-x^2/2\sigma^2} - \lim_{x\to+0} x^4 e^{-x^2/2\sigma^2} \\
&= \lim_{x\to+\infty} -x^4 e^{-x^2/2\sigma^2}
\end{aligned} \tag{28}$$

Based on the L'Hôpital's rule, the undetermined limit becomes:

$$\begin{aligned}\lim_{x\to+\infty} -x^4 e^{-x^2/2\sigma^2} &= -\lim_{x\to+\infty} \frac{x^4}{e^{x^2/2\sigma^2}} \\ &= \ ... \\ &= -\lim_{x\to+\infty} \frac{24}{(1/\sigma^2)(P(x)e^{x^2/2\sigma^2})} \\ &= 0.\end{aligned} \tag{29}$$

With $P(x)$ is polynomial and has a limit to $+\infty$. The second term is calculated in a same way (integration by parts) and $\mathbb{E}(|\mathbf{W}^Q|^2)$ becomes from Eq. (27):

$$\begin{aligned}\mathbb{E}(|\mathbf{W}^Q|)^2 &= \frac{1}{2\sigma^2}\int_0^\infty 4x^3 e^{-x^2/2\sigma^2}\,\mathrm{d}x \\ &= \frac{2}{\sigma^2}\left(x^2\sigma^2 e^{-x^2/2\sigma^2}\Big|_0^\infty + \int_0^\infty \sigma^2 2x e^{-x^2/2\sigma^2}\,\mathrm{d}x\right).\end{aligned} \tag{30}$$

The limit of first term is equals to 0 with the same method than in Eq. (29). Therefore, the expectation is:

$$\begin{aligned}\mathbb{E}(|\mathbf{W}^Q|^2) &= 4\left(\int_0^\infty x e^{-x^2/2\sigma^2}\,\mathrm{d}x\right) \\ &= 4\sigma^2.\end{aligned} \tag{31}$$

And finally, the variance is:

$$Var(|\mathbf{W}^Q|) = 4\sigma^2. \tag{32}$$

By analyzing their probability distribution, we demonstrate that the degrees of freedom for quaternion weights in encoders are four times higher than those of conventional graph encoder weights, providing enhanced representational capacity.

References

[Demsar2006] Janez Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[Glorot and Bengio2010] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of 13th International Conference on Automation-Intelligence-Safety*, volume 9, pages 249–256, 2010.

[He *et al.*2015] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the 2015 IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

[Parcollet *et al.*2019] Titouan Parcollet, Mirco Ravanelli, Mohamed Morchid, Georges Linarès, Chiheb Trabelsi, Renato De Mori, and Yoshua Bengio. Quaternion recurrent neural networks. In *Proceedings of 7th International Conference on Learning Representations*, 2019.

[Zhou *et al.*2022] Sheng Zhou, Hongjia Xu, Zhuonan Zheng, Jiawei Chen, Zhao Li, Jiajun Bu, Jia Wu, Xin Wang, Wenwu Zhu, and Martin Ester. A comprehensive survey on deep clustering: Taxonomy, challenges, and future directions. *CoRR*, abs/2206.07579, 2022.

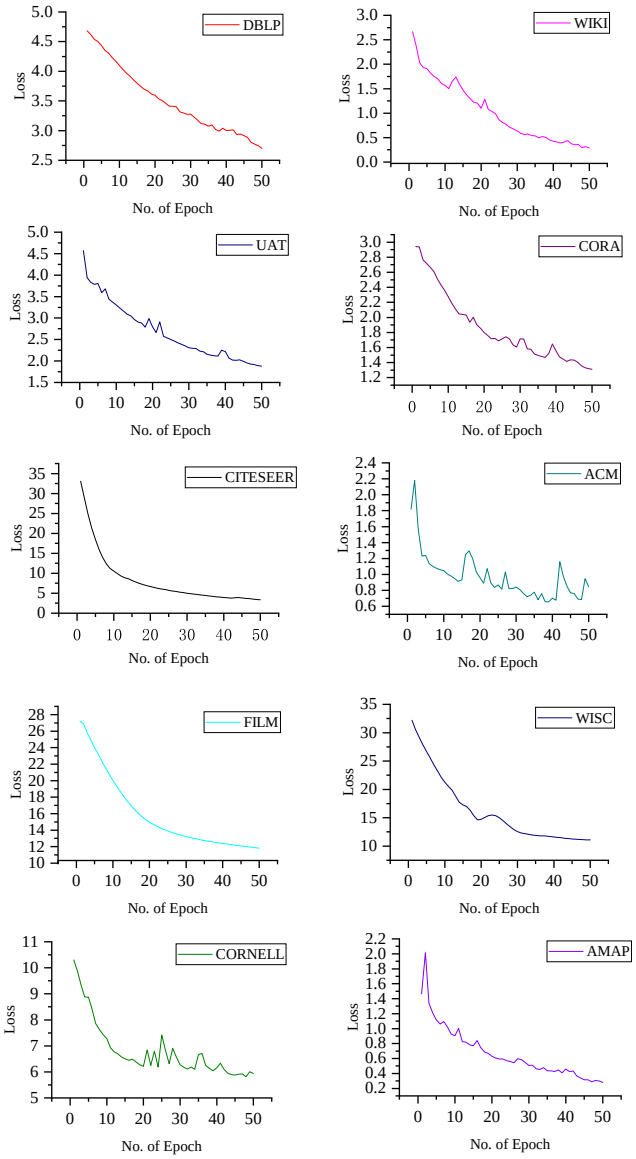

Fig. A.2. Convergence curves of the HyReaL on ten datasets.

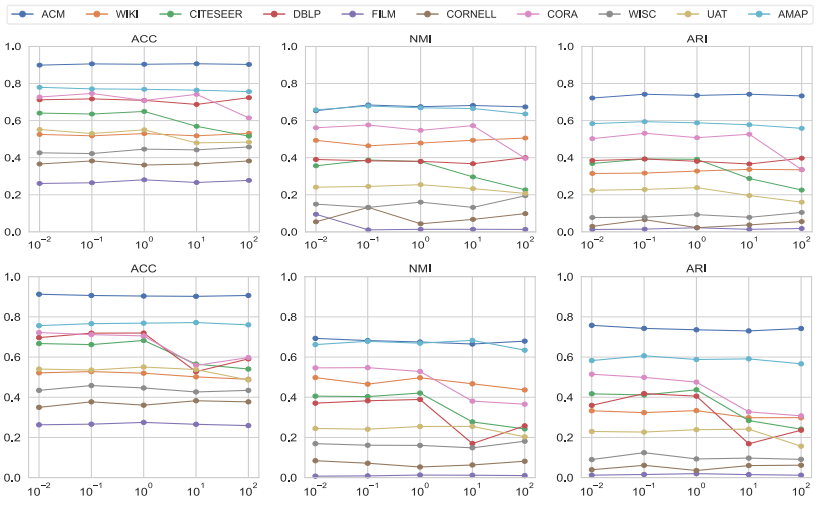

Fig. A.3. Sensitivity analysis of the trade-off parameters of the loss terms, i.e., α for $\mathcal{L}_{reg}$ (the upper row) and β for $\mathcal{L}_{sc}$ (the lower row), on all the ten datasets (marked in lines with different colors). x-axes indicate the values of α and β.

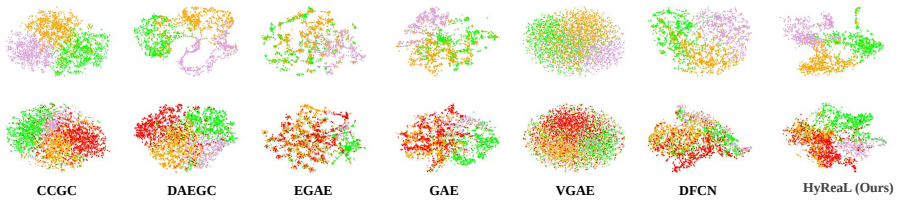

Fig. A.4. t-SNE visualization on ACM datasets. The first and second rows correspond to ACM and DBLP, respectively.

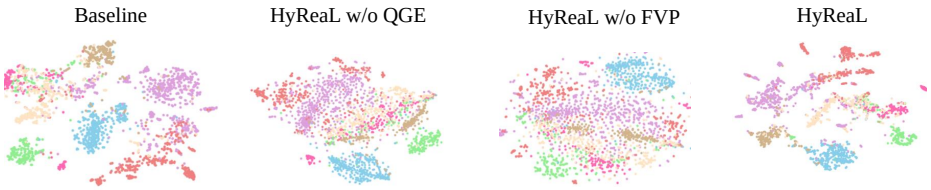

Fig. A.5. t-SNE visualization of the ablated variants of HyReaL on CORA dataset.

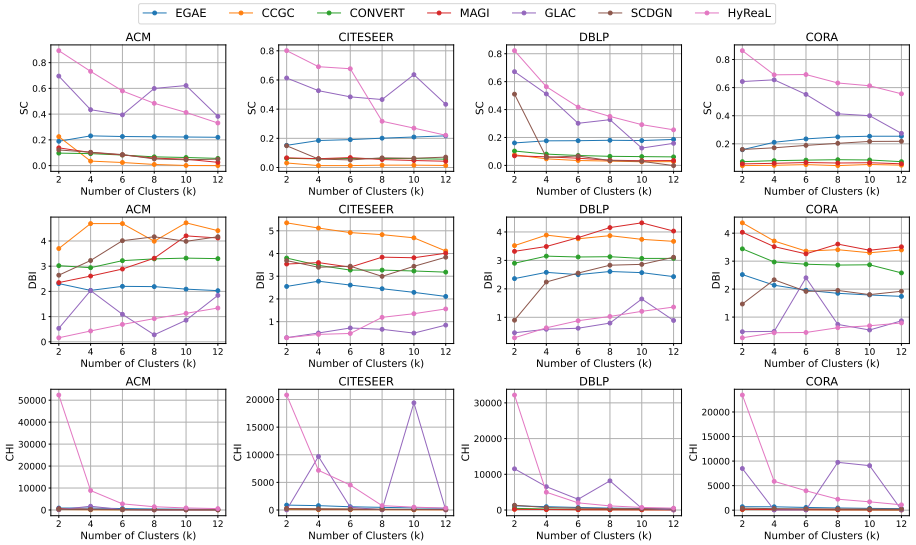

Fig. A.6. Clustering performance comparison using internal metrics under different ks. For the SC and CHI metrics, the higher the better. For the DBI metric, the lower the better.

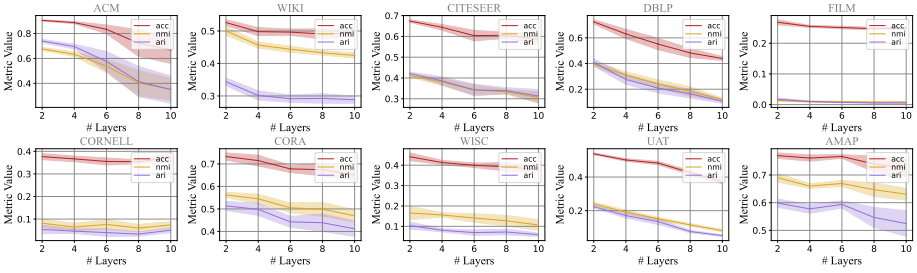

Fig. A.7. Supplementary Results of Over-smoothing Experiments.