A Hubness Perspective on Representation Learning for Graph-Based Multi-View Clustering

Supplementary Material

1. Introduction

In this supplementary, we provide the definitions of the hubness metrics used and provide quantitative analysis for multi-view datasets. Further, we provide the derivation of our view-specific hub-aware feature embedding, the pseudo-code, and an analysis of the complexity of hubREP. We also give experimental details and analysis, including hyperparameters, training time, and convergence analyses. Finally, we briefly discuss the limitations and future research directions as well as the social impact of our work.

Disclaimer: References in this supplementary material to figures/equations/tables in the main manuscript are denoted in cyan, while references to the supplementary are denoted in red.

2. Hubness Quantification in Multi-View Clustering

Hubness evaluation metrics. In this section, we briefly introduce two common metrics of hubness, *i.e.*, skewness (SK) [13] and hub occurrence (HO) [6]. Concretely, SK measures asymmetry in the k-occurrences distribution, with high values indicating a more severe hubness problem and vice versa. Given the k-occurrence score $N_k(x)$ for sample x, SK is calculated as

$$\mathrm{SK} = \frac{1}{\sigma_{N_k}^3}\mathbb{E}\left[\left(N_k(x) - \mu_{N_k}\right)^3\right], \tag{1}$$

where μ_{N_k} and σ_{N_k} denote the mean and standard deviation of the k-occurrences distribution, respectively.

Hub occurrence (HO) indicates the number of hubs, and is defined as

$$\mathrm{HO} = \frac{1}{k}\frac{1}{|H|}\sum_{x\in H} N_k(x), \tag{2}$$

where $H = \{h \in \mathbf{X} \mid N_k(x) \geq 2k\}$. Here, $\mathbf{X}$ denotes data samples.

The prevalence of the hubness problem in MVC. In Table 1, we present statistics of the hubness score of the raw data over eight datasets, demonstrating the consistent prevalence of the hubness problem also in the raw data. Note that this paper mainly focuses on the hubness problem within latent space, which has been analyzed in Sec. 3 of the main paper and further in Sec. 4 of this supplementary material.

3. Method

Calculation of p_{ij}^v and κ_i^v. Recall that $p_{ij}^v = \frac{1}{2}(p_{i|j}^v + p_{j|i}^v)$, where

$$p_{i|j}^v = \frac{\exp\left(\kappa_i^v({x_i^v}^\top x_j^v)/\|x_i^v\|\|x_j^v\|\right)}{\sum_{b,m}\exp\left(\kappa_i^v({x_b^v}^\top x_m^v)/\|x_b^v\|\|x_m^v\|\right)}. \tag{3}$$

Given $\mathbf{X}^v = \{x_i^v\}_{i=1}^N$ as input, we follow the approach in [18] and employ a binary search to find κ_i^v that produces P_i^v with a fixed perplexity P:

$$|\log_2(P) - H(P_i^v)| \leq 0.1 \cdot \log_2(P), \tag{4}$$

in which $H(P_i^v)$ represents the Shannon entropy of P_i^v and can be calculated as

$$H(P_i^v) = \sum_{j=1}^{n} p_{i|j}^v \log_2(p_{i|j}^v) \tag{5}$$

Complete derivation of the view-specific hub-aware feature embedding. In this section, we give a detailed derivation of the view-specific hub-aware feature embedding, where the objective for our view-specific embedding $f^v : \mathcal{X}^v \to \mathcal{H}^v$ is defined as

$$\arg\min_{\Theta^v} KL(P^v||Q^v) = \arg\min_{\Theta^v}\sum_{i,j} p_{ij}^v \log\frac{p_{ij}^v}{q_{ij}^v}, \tag{6}$$

where p_{ij} and q_{ij} are pairwise similarities in the input and latent space. Note that the optimization of $KL(P^v||Q^v)$ is similar to [16], with one difference being that we optimize over the parameter Θ^v. Denote $\mathcal{L}_s^v = KL(P^v||Q^v)$, then we have

$$\arg\min_{\Theta^v}\mathcal{L}_s^v = \arg\min_{\Theta^v}\sum_{i,j} p_{ij}^v \log\frac{p_{ij}^v}{q_{ij}} \tag{7}$$

$$= \arg\min_{\Theta^v}\sum_{i,j} p_{ij}^v \log p_{ij}^v - \sum_{i,j} p_{ij}\log q_{ij}^v \tag{8}$$

$$= \arg\min_{\Theta^v} -\sum_{i,j} p_{ij}^v \log q_{ij}^v, \tag{9}$$

where term $\sum_{i,j} p_{ij}^v \log p_{ij}^v$ can be neglect as it is constant *w.r.t* Θ^v. Recall that the definition of q_{ij}^v is given by

$$q_{ij}^v = \frac{\exp\left(\kappa f^v(x_i^v|\Theta^v)^\top f^v(x_j^v|\Theta^v)\right)}{\sum_{b,m}\exp\left(\kappa f^v(x_b^v|\Theta^v)^\top f^v(x_m^v|\Theta^v)\right)}. \tag{10}$$

Dataset	View 1		View 2		View 3		View 4		View 5		AVG	
	SK	HO	SK	HO	SK	HO	SK	HO	SK	HO	SK	HO
3Sources	5.4669	0.7799	5.7079	0.8355	5.8571	0.8296	-	-	-	-	5.6773	0.8150
BBCSport	10.6746	0.8324	10.5150	0.8798	-	-	-	-	-	-	10.5948	0.8561
BDGP	2.4178	0.4929	1.5652	7.6032	-	-	-	-	-	-	1.9915	4.0480
MNIST-USPS	0.8484	0.2569	0.9693	0.2460	-		-	-	-	-	0.9089	0.2514
Hdigit	0.9151	0.2656	0.9956	0.2573	-	-	-	-	-	-	0.9553	0.2615
Reuters	16.5994	0.9112	16.6068	0.9137	16.5078	0.9097	15.9340	0.9090	16.5582	0.9063	16.4412	0.9100
Animal	14.9773	0.7082	19.4311	0.6508	5.4516	0.7170	8.4705	0.7630	-	-	12.0826	0.7098
Noisymnist	0.9343	0.2859	4.4729	0.5530	-	-	-	-	-	-	2.7036	0.4194

Table 1. SK and HO for raw data in different datasets

Substituting Eq. 10 into Eq. 9, we get:

$$\arg\min_{\Theta^v} -\sum_{i,j} p_{ij}^v \log q_{ij}^v \tag{11}$$

$$= \arg\min_{\Theta^v} \Bigg(-\sum_{i,j} p_{ij}^v \big(\kappa f^v(x_i^v|\Theta^v)^\top f^v(x_j^v|\Theta^v)\big) + \sum_{i,j} p_{ij}^v \log\Big(\sum_{b,m} \exp\big(\kappa f^v(x_b^v|\Theta^v)^\top f^v(x_m^v|\Theta^v)\big)\Big)\Bigg) \tag{12}$$

$$= \arg\min_{\Theta^v} \Bigg(-\sum_{i,j} p_{ij}^v \kappa f^v(x_i^v|\Theta^v)^\top f^v(x_j^v|\Theta^v) + \underbrace{\Big(\sum_{i,j} p_{ij}^v\Big)}_{=1} \Big(\log \sum_{b,m} \exp\big(\kappa f^v(x_b^v|\Theta^v)^\top f^v(x_m^v|\Theta^v)\big)\Big)\Bigg) \tag{13}$$

$$= \arg\min_{\Theta^v} \Bigg(\underbrace{-\sum_{i,j} p_{ij}^v \kappa f^v(x_i^v|\Theta^v)^\top f^v(x_j^v|\Theta^v)}_{\text{Neighborhood Preservation}} + \underbrace{\log \sum_{b,m} \exp\big(\kappa f^v(x_b^v|\Theta^v)^\top f^v(x_m^v|\Theta^v)\big)}_{\text{Hubness Reduction}}\Bigg) \tag{14}$$

$$\tag{15}$$

Pseudo Code. The pseudo-code of the complete learning procedure of hubREP is summarized in Algorithm 1. Note that the equation number refers to equations in the main paper.

Complexity Analysis. In Algorithm 1, the primary computational operations include the calculation of Eq. 2, Eq. 5, Eq. 7, the graph generation process and the message aggregation inside each graph neural network layer. Specifically, the corresponding κ_i in Eq. 2 is calculated by binary search, which generally takes $\mathcal{O}(nlog(R))$, where R denotes the range of the search. The calculation of Eq. 5 and Eq. 7 has

Algorithm 1 The **hubREP** Algorithm

Input: Multi-view data $\{\mathbf{X}^v\}_{v=1}^V$, number of clusters C, nearest neighbor k, trade-off parameters α, β and λ, hyper-parameter κ, total training epoch T.
Output: The clustering result $\bar{\mathbf{Y}}$.

while epoch $\leq T$ **do**
  Obtain $\{p_{ij}^v\}_{v=1}^V$ for each view by Eq. 2.
  Calculate view-specific hub-aware embedding loss $\mathcal{L}_s$ by Eq. 5.
  Calculate hub-aware cross-view pairing loss $\mathcal{L}_c$ by Eq. 7.
  Calculate graph autoencoder reconstruction loss $\mathcal{L}_{grec}$ by Eq. 9.
  Update the parameters of the whole model by minimizing Eq. 10.
end while
Return: the consensus representation.
Perform Kmeans algorithm on the consensus representation to output the clustering result $\bar{\mathbf{Y}}$.

a computational complexity of $\mathcal{O}(VN^2d)$ and $\mathcal{O}(V^2N^2d)$, respectively, where V indicates the number of views and d denotes the latent space dimension. The computational complexity for graph building and each GNN layer is approximate $\mathcal{O}(VN^2d)$ and $\mathcal{O}(Nkd_gd_g^{'})$ respectively, with d_g and $d_g^{'}$ representing the input and output dimensions of each GNN layer.

Discussion on alternative local structure preservation. Comapared to other existing local structure preservation approaches, our motivation lies in considering hubness and neighborhood information jointly. Preserving structure only will suffer or even amplify the hubness problem, potentially collapsing representations (*i.e.* $\lambda = 0$). Note, $\mathcal{L}_{np}$, which naturally arises as part of our formulation (Eq. 4), can be recognized as the loss function in Laplacian Eigenmaps [2]. This potentially provides opportunities to exploit alternative graph-based regularization techniques either by considering alternative formulations of the problem or through ad-hoc

substitutions of this loss.

Relation to works that utilize hubs for clustering. In traditional single-view clustering methods [7, 14] and very recently non-deep multi-view clustering [26], hubness has been discussed from a positive perspective, where hubs are selected as cluster centroids/anchors. We instead, through a thorough analysis, demonstrate the problem of hubness especially when considering a representation learning (deep) setting, where representations are used to construct graphs for message passing in GMVC.

4. Motivation Experiments

Implementation details. In our motivation experiments, we employ view-specific autoencoders with reconstruction loss applied to the input data for each view. The shared representation is combined by averaging all view-specific embeddings. We pre-train each autoencoder for 300 epochs, followed by 200 epochs of finetuning both the autoencoder and graph autoencoders jointly. The Adam optimizer is employed with a learning rate of 0.0005. For k-nearest neighbor graph construction and hubness score evaluation, we set the hyperparameter k as 5. The dimensionality of the latent space embedding is set to 1024 unless otherwise stated.

Significance of hub-aware cross-view pairing. In Figure 1, we plot the absolute difference of k-occurrence score $N_k(x^v)$ for two different views on BDGP and 3Sources. We can consistently observe that the hubness remains different for samples from different views, emphasizing the significance of considering cross-view hubness to ensure inter-view consistency.

Full results for *Observation 3: Reducing hubness enhances GNN-based MVC*. In Table 2, we provide the full results for the hubness reduction technique experiments in the motivation section (Sec. 3) of the main paper, covering eight datasets with six clustering metrics and two hubness metrics. Further statistical analysis over these datasets, *i.e.*, Pearson correlation analysis is presented in Figure 2, which supplement the results in Figure 2e and 2f of the main paper. We observe a persistent negative correlation between clustering performance metrics (ACC, NMI, and ARI) and hubness scores (SK and HO), convincingly validating that reducing hubness techniques could benefit the performance of GNN-based MVC.

5. Experiments

5.1. Setup

Implementation details. In this paper, all our experiments are conducted on the PyTorch platform, running on a Linux server equipped with an Intel(R) Xeon(R) Gold 5218R CPU @ 2.10GHz and an NVIDIA GeForce RTX 3090 GPU. For each view-specific encoder, the dimension of each layer is set to [d_v,2000,500,500,1024], where d_v denotes the dimension of the representation in each view. For each view-specific graph convolutional encoder, the dimension is set to [512, 2048, 256], while the dimensions of the decoder in the graph autoencoder are reversed. We train the model for 300 epochs with Adam optimizer. The learning rate is set to 0.0001. We recommend κ from [0.1,0.5] and λ from [0.3,0.7]. When computing κ_i^v, the perplexity P is set as 45. During training, graphs are dynamically constructed at each epoch. To speed up training, we accelerate the kNN search using Faiss [8], which is faster than the scikit-learn package (*e.g.*, 0.45s vs 0.02s for BBCSport with 544 samples and 180s vs 7.5s for Animal with 11,673 samples).

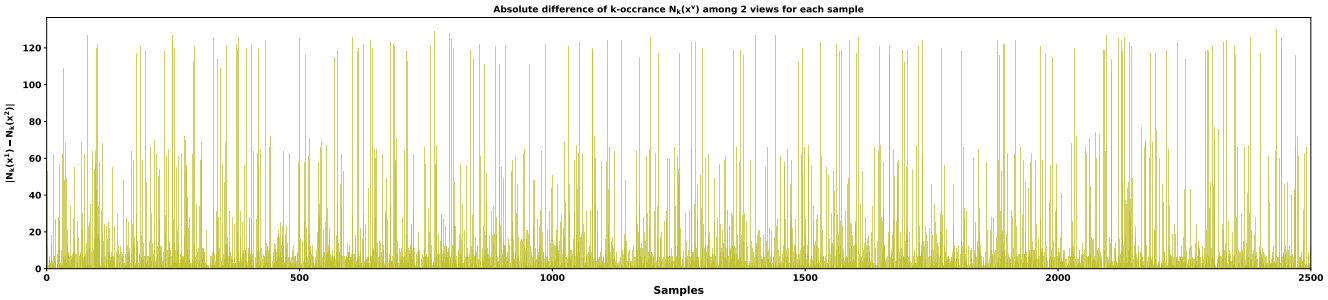
Datasets. In this study, we employ eight common multi-view datasets to evaluate the performance of hubREP, the details are described as follows.

- **3Sources** [1] consists of 169 news stories from BBC, Reuters, and The Guardian, each serving as a unique view. The articles are classified into six categories: business, entertainment, health, politics, sport, and technology.
- **BBCSport** [2] is a two-view dataset consisting of 544 documents from the BBC Sport website across five sports topics: athletics, cricket, football, rugby, and tennis, which denote five different classes.
- **BDGP** [3], short for the Berkeley Drosophila Genome Project, consists of 2,500 drosophila embryo samples across five categories, with visual and textual features representing two distinct views.
- **MNIST-USPS** [12] contains 5,000 handwritten digit images across ten categories, derived from two sub-datasets treated as two unique views.
- **Hdigit** [4] comprises two types of heterogeneous representations from 10,000 handwritten digit samples across ten classes.
- **Reuters** [1] includes 1,200 articles across six categories, with each article written in five languages.
- **Animals** [10] consists of 11,673 images across 20 categories, characterized by four heterogeneous feature representations.
- **Noisymnit** [20] is a two-view database containing 70,000 handwritten digit samples across ten classes. For this study, we select a subset containing 15,000 samples.

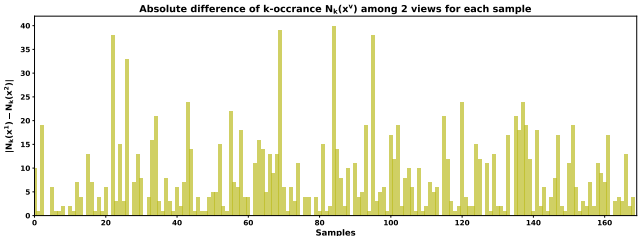
Baselines. We evaluate our methods against ten state-of-the-art approaches with details as follows.

- **MCGC** [27] is a traditional graph-based method that learns a consensus graph by minimizing inter-view disagreement and constraining the rank of the Laplacian matrix to obtain k-cluster assignments without post-processing directly.
- **EAMC** [28] introduces an end-to-end framework that combines adversarial learning and attention mechanisms to align latent feature distributions and evaluate the signifi-

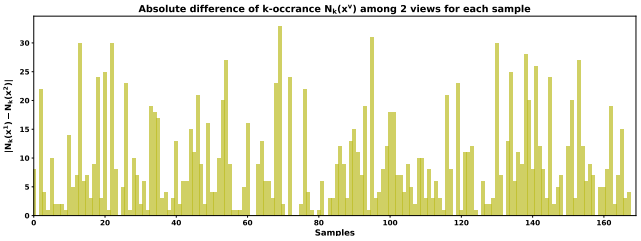
[1] http://mlg.ucd.ie/datasets/3sources.html
[2] http://mlg.ucd.ie/datasets/segment.html



(a) BDGP



(b) 3Sources view1 vs. view2



(c) 3Sources view1 vs. view3

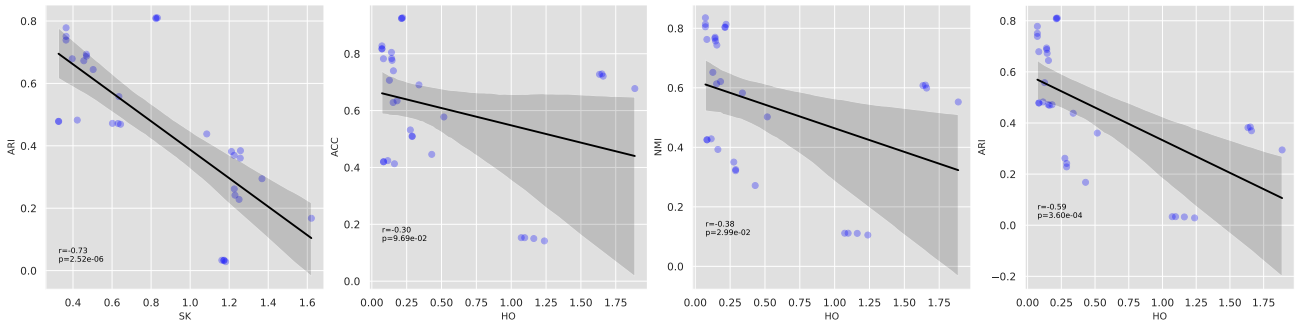
Figure 1. Visulzaition of absolute difference of $N_k(x^v)$ across two different views on BDGP and 3Sources dataset.

	MNIST-USPS				3Sources				Animal				Noisymnist			
Method	None	L2	ZN	CL2N	None	L2	ZN	CL2N	None	L2	ZN	CL2N	None	L2	ZN	CL2N
ACC	0.7400	0.8048	0.7834	0.7766	0.5775	0.7065	0.6284	0.6343	0.1415	0.1529	0.1499	0.1530	0.4130	0.4240	0.4201	0.4198
NMI	0.7436	0.7657	0.7699	0.7574	0.5023	0.6519	0.6134	0.6211	0.1052	0.1114	0.1108	0.1113	0.3928	0.4285	0.4253	0.4251
ARI	0.6445	0.6935	0.6875	0.6726	0.3605	0.5579	0.4719	0.4716	0.0286	0.0334	0.0324	0.0333	0.4690	0.4824	0.4784	0.4782
PUR	0.7858	0.8067	0.8110	0.7937	0.7089	0.8201	0.7692	0.7811	0.1910	0.2043	0.2029	0.2034	0.2617	0.2821	0.2779	0.2776
PRE	0.6823	0.7264	0.7237	0.7076	0.6014	0.7241	0.6773	0.6832	0.0929	0.0977	0.0973	0.0978	0.3574	0.3853	0.3833	0.3831
F1	0.7071	0.7315	0.7377	0.7197	0.5727	0.6797	0.6213	0.6370	0.1236	0.1209	0.1156	0.1157	0.3483	0.3731	0.3706	0.3704
SK	0.5032	0.4684	0.4695	0.4553	1.2567	0.6351	0.6019	0.6306	1.1817	1.1736	1.1725	1.1622	0.6436	0.4219	0.3257	0.3268
HO	0.1547	0.1408	0.1421	0.1452	0.5168	0.1259	0.1527	0.1813	1.2379	1.0968	1.1620	1.0720	0.1621	0.1142	0.0835	0.0843
	BDGP				BBCSport				Hdigit				Reuters			
Method	None	L2	ZN	CL2N	None	L2	ZN	CL2N	None	L2	ZN	CL2N	None	L2	ZN	CL2N
ACC	0.6775	0.7211	0.7275	0.7290	0.6904	0.9246	0.9239	0.9265	0.7824	0.8176	0.8166	0.8276	0.4460	0.5102	0.5320	0.5087
NMI	0.5524	0.5991	0.6076	0.6091	0.5826	0.8045	0.8027	0.8133	0.7626	0.8050	0.8137	0.8354	0.2716	0.3252	0.3504	0.3214
ARI	0.2946	0.3695	0.3818	0.3844	0.4381	0.8104	0.8076	0.8101	0.6790	0.7388	0.7508	0.7783	0.1678	0.2417	0.2619	0.2282
PUR	0.6778	0.7214	0.7275	0.7290	0.7809	0.9511	0.9496	0.9467	0.7970	0.8324	0.8327	0.8530	0.5017	0.5418	0.5507	0.5302
PRE	0.6052	0.6474	0.6543	0.6557	0.6724	0.8622	0.8607	0.8644	0.7226	0.7702	0.7793	0.8036	0.3321	0.3832	0.3990	0.3770
F1	0.6144	0.6480	0.6541	0.6552	0.6431	0.8612	0.8599	0.8643	0.7290	0.7838	0.7940	0.8257	0.3744	0.4067	0.4257	0.4098
SK	1.3679	1.2256	1.2117	1.2577	1.0851	0.8247	0.8252	0.8327	0.3959	0.3642	0.3645	0.3645	1.6209	1.2296	1.2261	1.2506
HO	1.8869	1.6597	1.6336	1.6511	0.3381	0.2135	0.2131	0.2208	0.0831	0.0726	0.0725	0.0725	0.4298	0.2898	0.2762	0.2891

Table 2. Full results of the motivation experiment, including the clustering performance and hub score for various hubness reduction techniques across eight datasets (k=5).

cance of each modality.

- **MFLVC** [23] concentrates on deep subspace learning by employing contrastive regularizations within multi-level feature spaces to learn shared semantics across multi-view data.
- **AECoDDC** [17] is a suggested instance of DeepMVC that employs NT-Xent contrastive loss [15] for view-specific and multi-view self-supervised learning, subsequently integrating a weighted fusion and DDC clustering module [9].
- **GCFAgg** [25] proposes a plug-in module that focuses on global and cross-view representation aggregation through leveraging transformer mechanisms and structure-guided contrastive learning.
- **DFP-GNN** [22] integrates Graph Neural Network-based view-specific and consensus graph representation learning, capturing consistency and complementarity across many views and further enhancing clustering performance.
- **DMCAG** [5] builds anchor graphs within latent spaces and leverages spectral self-supervised learning to generate pseudo-labels for directly optimizing view-specific feature embeddings, thus enhancing clustering performance and efficacy.
- **S^{2}MVTC** [11] constructs and rotates a tensor and in-



(a) (b) (c) (d)

Figure 2. Pearson correlation coefficient between clustering performance (ACC, NMI and ARI) and hubness score (SK and HO).

	MNIST-USPS					3Sources					Animal					Noisymnist				
Method	None	L2	ZN	CL2N	Ours	None	L2	ZN	CL2N	Ours	None	L2	ZN	CL2N	Ours	None	L2	ZN	CL2N	Ours
ACC	0.7766	0.8042	0.7663	0.7855	**0.9572**	0.5609	0.6118	0.6055	0.6213	**0.9290**	0.1442	0.1591	0.1635	0.1625	**0.2966**	0.3997	0.4243	0.4207	0.4213	**0.6815**
NMI	0.7680	0.7769	0.7624	0.7719	**0.9090**	0.4848	0.6078	0.6121	0.6036	**0.8246**	0.1079	0.1156	0.1155	0.1187	**0.4904**	0.4241	0.4502	0.4436	0.4448	**0.6802**
ARI	0.6793	0.7038	0.6745	0.6905	**0.9084**	0.3578	0.4553	0.4665	0.4503	**0.8519**	0.0302	0.0358	0.0370	0.0374	**0.2071**	0.2731	0.2961	0.2906	0.2909	**0.5785**
PUR	0.8023	0.8221	0.7954	0.8125	**0.9592**	0.6864	0.7408	0.7830	0.7613	**0.9408**	0.1972	0.2088	0.2030	0.2162	**0.3590**	0.4832	0.4989	0.4944	0.4940	**0.6898**
PRE	0.7152	0.7355	0.7121	0.7245	**0.9191**	0.5864	0.6885	0.6936	0.6744	**0.8692**	0.0947	0.1003	0.1009	0.1016	**0.2334**	0.3789	0.4029	0.3987	0.3994	**0.6205**
F1	0.7332	0.7482	0.7302	0.7386	**0.9190**	0.5459	0.6273	0.6236	0.6134	**0.8712**	0.1249	0.1190	0.1216	0.1195	**0.2334**	0.3708	0.3904	0.3856	0.3863	**0.6220**
SK	0.4052	0.3499	0.3460	0.3501	**0.1464**	1.3570	0.4736	0.4755	0.5396	**0.3931**	1.0408	1.0297	1.0152	1.0224	**0.0924**	0.5981	0.4774	0.3589	0.3586	**0.2344**
HO	0.0794	0.0669	0.0680	0.0669	**0.0647**	0.5148	0.0654	0.0662	0.0675	**0.0542**	0.6746	0.5952	0.5448	0.5773	**0.0464**	0.1069	0.0991	0.0673	0.0674	**0.0528**
	BDGP					BBCSport					Hdigit					Reuters				
Method	None	L2	ZN	CL2N	Ours	None	L2	ZN	CL2N	Ours	None	L2	ZN	CL2N	Ours	None	L2	ZN	CL2N	Ours
ACC	0.6938	0.7738	0.7671	0.7713	**0.9868**	0.7210	0.9161	0.9298	0.9283	**0.9632**	0.7621	0.8328	0.8333	0.8320	**0.9661**	0.4265	0.5139	0.4898	0.4913	**0.6533**
NMI	0.5984	0.6580	0.6501	0.6555	**0.9538**	0.6301	0.7895	0.8136	0.8109	**0.8948**	0.7531	0.8422	0.8426	0.7985	**0.9331**	0.2597	0.3295	0.3147	0.3162	**0.4330**
ARI	0.3683	0.4747	0.4603	0.4699	**0.9674**	0.5070	0.7967	0.8200	0.8156	**0.9070**	0.6540	0.7869	0.7876	0.7396	**0.9295**	0.1613	0.2402	0.2190	0.2242	**0.3628**
PUR	0.7195	0.7744	0.7678	0.7719	**0.9916**	0.7989	0.9430	0.9507	0.9489	**0.9853**	0.7752	0.8556	0.8561	0.8421	**0.9662**	0.4853	0.5450	0.5148	0.5267	**0.6567**
PRE	0.6414	0.7009	0.6938	0.6984	**0.9740**	0.7034	0.8495	0.8709	0.8684	**0.9298**	0.6978	0.8108	0.8112	0.7692	**0.9370**	0.3198	0.3809	0.3646	0.3680	**0.4921**
F1	0.6516	0.6965	0.6896	0.6943	**0.9740**	0.6842	0.8475	0.8700	0.8675	**0.9297**	0.7096	0.8386	0.8390	0.7791	**0.9377**	0.3714	0.4167	0.4110	0.4117	**0.4946**
SK	1.1626	1.1321	1.0841	1.0006	**0.8746**	1.0121	0.7101	0.6978	0.6981	**0.1463**	0.4472	0.3804	0.3810	0.3810	**0.1190**	1.5529	1.1722	1.1010	1.1010	**0.6298**
HO	0.9647	0.9583	0.9545	0.9364	**0.8946**	0.2849	0.1455	0.1480	0.1474	**0.0884**	0.0891	0.0798	0.0798	0.0798	**0.0389**	0.3931	0.2184	0.2150	0.2150	**0.2046**

Table 3. Comparison of clustering performance and hub score of ours and different hubness reduction techniques on eight datasets (k=10).

troduces a tensor low-frequency approximation operator, tailored for anchor-based algorithms to learn the intra- and inter-view correlations among embeddings simultaneously.

- **MVCAN** [24] develops parameter-decoupled deep models and a dual-level iterative optimization technique to tackle noisy views in practical multi-view clustering. Note that the reproduction of MVCAN requires pre-defined anchor graphs as inputs, hence we utilize the approach [21] outlined in MVCAN for anchor graph preparation. The anchor number is set to 1000. For small datasets under 1000 instances, we utilize the entire dataset.
- **SURER** [19] refines view-specific graphs through graph neural networks and constructs a unified heterogeneous graph across all views to effectively integrate complementary information from multiple views for enhanced clustering performance.

5.2. Additional Experimental Results and Analysis

Clustering and hubness reduction performance. In Table 3, we present the full overview of the clustering performance and corresponding hubness scores in comparison to other hub reduction techniques, demonstrating the effectiveness of hubREP for simultaneously eliminating hubness while enhancing the clustering performance.

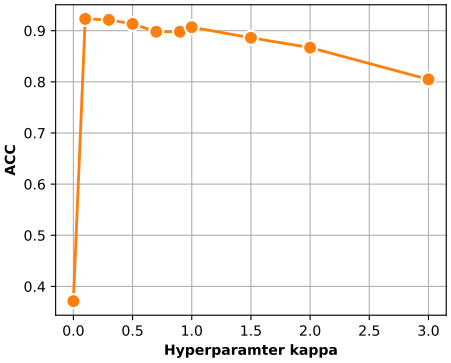


Figure 3. The effect of hyperparameter κ on clustering performance ACC on 3Sources dataset.

Effect of hyperparameter κ. In Figure 3, we illustrate the effects of κ from set $\{0, 0.1, 0.3, 0.5, 0.7, 0.9, 1, 1.5, 2, 3\}$ for ACC on 3Sources dataset. We recommend choosing $\kappa \in [0.1, 0.5]$ for better clustering performance.

Ablation study: effects of hub-aware cross-view pairing vs. equally weighted cross-view pairing. In Table 4, we

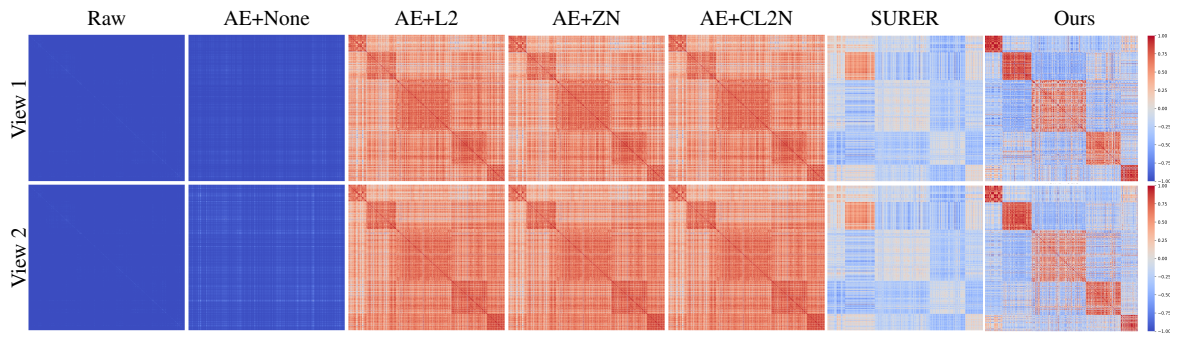


Figure 4. The inner product of representations embedded by different methods on the BBCSport dataset for each view.

	Method	ACC	NMI	ARI	PUR
3Sources	hubREP	**0.9290**	**0.8246**	**0.8519**	**0.9408**
	hubREP-E	0.9231	0.8191	0.8406	0.9231
BBCSport	hubREP	**0.9632**	**0.8948**	**0.9070**	**0.9853**
	hubREP-E	0.9416	0.8400	0.8520	0.9573
BDGP	hubREP	**0.9868**	**0.9538**	**0.9674**	**0.9916**
	hubREP-E	0.9584	0.8773	0.8995	0.9692
Reuters	hubREP	**0.6533**	**0.4330**	**0.3628**	**0.6567**
	hubREP-E	0.5844	0.3754	0.3024	0.6085

Table 4. Effects of equally cross-view paring and hub-aware cross-view paring

present the results for our proposed method, hubREP, alongside an ablated variant, hubREP-E. In hubREP-E, the hub-aware cross-view pairing module is replaced with an equally weighted cross-view pairing module, in which $\mathbf{S}^u\mathbf{S}^{v\top}$ in Eq. 7 in the main paper is removed. The results show that hubREP consistently outperforms hubREP-E across four datasets, highlighting the effectiveness of hub-aware weighting during cross-view alignment over a uniform alignment strategy.

Training times. In Table 5, we present the training time for each method on four datasets. It can be observed that our method ranks third in training time, primarily due to the dynamic graph construction in each epoch, while SURER and DFP-GNN prepare graphs before training. DMCAG exhibits the longest training time compared to the other methods, which mainly lies in its convex quadratic programming for deriving the anchor graph matrix for each view during the fusion process.

Reconstuction ablation. Denoting attribute and graph reconstruction as Attr and Rec, respectively, we conduct an ablation study to verify their effect (see Table 6).

Visualization. Due to space limitation in the main paper, in Figure 4, we present the full comparison of the inner product of the different representations obtained from: raw data, conventional normalization methods (including AE+None, AE+ZN, and AE+CL2N), SURER [19] and our method. Our method obtains clearer structural alignment and superior clustering performance.

	DFPGNN	SURER	DMCAG	Ours
3Sources	1.51×10^{-2}	1.96×10^{-2}	1.81×10^{1}	9.07×10^{-1}
BBCSport	1.22×10^{-2}	2.14×10^{-2}	1.44×10^{1}	2.84×10^{0}
BDGP	1.89×10^{-2}	1.25×10^{-1}	6.87×10^{1}	4.35×10^{0}
Reuters	2.45×10^{-2}	1.47×10^{-1}	9.18×10^{1}	6.36×10^{0}

Table 5. Training time (seconds per epoch) of four methods on four datasets.

ACC	3Sources	BBCSport	BDGP	Reuter
-Attr	0.905	0.941	0.938	0.620
-Rec	0.876	0.932	0.959	0.614
-Attr-Rec	0.828	0.927	0.925	0.587
hubREP	**0.929**	**0.963**	**0.987**	**0.653**

Table 6. Reconstuction ablation experiment.

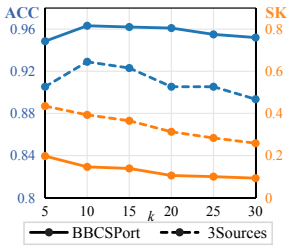
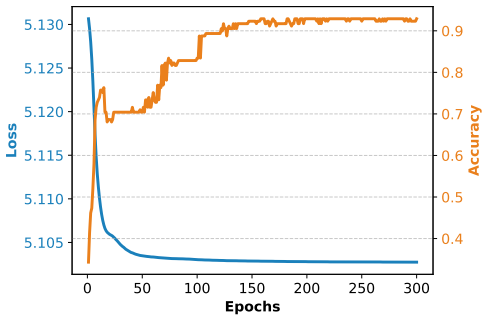


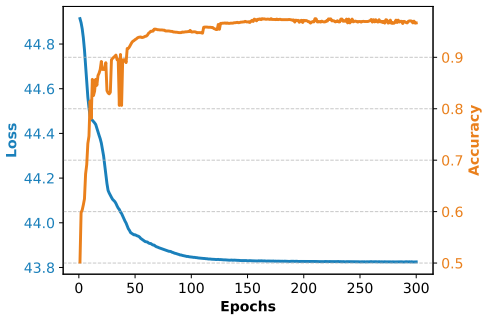
Figure 5. Sensitivity analysis on k in kNN.

Sensitivity analysis We analyze the effect of k in the kNN in Figure 5. While relatively robust, very small or large k degrade performance. As k increases, SK declines, reflecting reduced dominance of hubs in neighborhoods and inclusion of less-relevant neighbors.

Convergence analysis. In Figure 6, we present the total loss value and the accuracy metrics against epochs on the 3Sources and BDGP datasets. We can observe that the proposed hubREP exhibits good convergence properties, as shown by the monotonically decreasing loss value and a



(a) 3Sources



(b) BDGP

Figure 6. Convergence curves on 3Sources and BDGP dataset.

steady rise in accuracy as the epoch increases, which eventually converges.

6. Limitations and Future Work

In this work, we focus on reducing hubness and encouraging consistency across the views. However, while we show that reducing hubness leads to improved performance, not all hubs are necessarily bad hubs. In future work, we will focus on obtaining reliable pseudo-cluster-labels to further analyze the impact of good hubs and bad hubs for multi-view clustering, potentially leveraging this knowledge to further refine the embedding process. Furthermore, considering the practical challenges of missing instances in real-world scenarios, another direction for future research lies in extending the proposed method to handle incomplete multi-view clustering effectively.

7. Potential Negative Societal Impacts

As with most methodological research, the proposed method could be applied to downstream tasks that may carry potential societal risks. Given the focus on unsupervised learning and the absence of labels in our work, it is particularly important to evaluate what the model has learned, especially when deployed in scenarios requiring high-stakes decisions.

References

[1] Massih R Amini, Nicolas Usunier, and Cyril Goutte. Learning from multiple partially observed views-an application to multilingual text categorization. *NIPS*, pages 28–36, 2009. 3

[2] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003. 2

[3] Xiao Cai, Hua Wang, Heng Huang, and Chris Ding. Joint stage recognition and anatomical annotation of drosophila gene expression patterns. *Bioinform.*, 28(12):16–24, 2012. 3

[4] Man-Sheng Chen, Jia-Qi Lin, Xiang-Long Li, Bao-Yu Liu, Chang-Dong Wang, Dong Huang, and Jian-Huang Lai. Representation learning in multi-view clustering: a literature review. *Data Sci. Eng.*, 7(3):225–241, 2022. 3

[5] Chenhang Cui, Yazhou Ren, Jingyu Pu, Xiaorong Pu, and Lifang He. Deep multi-view subspace clustering with anchor graph. In *IJCAI*, pages 3577–3585, 2023. 4

[6] Arthur Flexer and Dominik Schnitzer. Choosing p norms in high-dimensional spaces based on hub analysis. *Neurocomputing*, 169:281–287, 2015. 1

[7] Zhenfeng He. Hub selection for hub based clustering algorithms. In *2014 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 479–484. IEEE, 2014. 3

[8] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Trans. Big Data*, 7(3): 535–547, 2021. 3

[9] Michael Kampffmeyer, Sigurd Løkse, Filippo Maria Bianchi, Lorenzo Livi, Arnt-Børre Salberg, and Robert Jenssen. Deep divergence-based approach to clustering. *Neural Networks*, 113:91–101, 2019. 4

[10] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36 (3):453–465, 2014. 3

[11] Zhen Long, Qiyuan Wang, Yazhou Ren, Yipeng Liu, and Ce Zhu. S2mvtc: a simple yet efficient scalable multi-view tensor clustering. In *CVPR*, pages 24088–24097, 2024. 4

[12] Xi Peng, Zhenyu Huang, Jiancheng Lv, Hongyuan Zhu, and Joey Tianyi Zhou. Comic: Multi-view clustering without parameter selection. In *ICML*, pages 5092–5101, 2019. 3

[13] Milos Radovanovic, Alexandros Nanopoulos, and Mirjana Ivanovic. Hubs in space: Popular nearest neighbors in high-dimensional data. *J. Mach. Learn. Res.*, 11:2487–2531, 2010. 1

[14] Nenad Tomasev, Milos Radovanovic, Dunja Mladenic, and Mirjana Ivanovic. The role of hubness in clustering high-dimensional data. *TKDE*, 26(3):739–751, 2013. 3

[15] Daniel J Trosten, Sigurd Lokse, Robert Jenssen, and Michael Kampffmeyer. Reconsidering representation alignment for multi-view clustering. In *CVPR*, pages 1255–1265, 2021. 4

[16] Daniel J Trosten, Rwiddhi Chakraborty, Sigurd Løkse, Kristoffer Knutsen Wickstrøm, Robert Jenssen, and Michael C Kampffmeyer. Hubs and hyperspheres: Reducing hubness and improving transductive few-shot learning with hyperspherical embeddings. In *CVPR*, pages 7527–7536, 2023. 1

[17] Daniel J Trosten, Sigurd Løkse, Robert Jenssen, and Michael C Kampffmeyer. On the effects of self-supervision and contrastive alignment in deep multi-view clustering. In *CVPR*, pages 23976–23985, 2023. 4

[18] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *J. Mach. Learn. Res.*, 9(11), 2008. 1

[19] Jing Wang, Songhe Feng, Gengyu Lyu, and Jiazheng Yuan. Surer: Structure-adaptive unified graph neural network for multi-view clustering. In *AAAI*, pages 15520–15527, 2024. 5, 6

[20] Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. On deep multi-view representation learning. In *ICML*, pages 1083–1092, 2015. 3

[21] Wei Xia, Quanxue Gao, Qianqian Wang, Xinbo Gao, Chris Ding, and Dacheng Tao. Tensorized bipartite graph learning for multi-view clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(4):5187–5202, 2023. 5

[22] Shunxin Xiao, Shide Du, Zhaoliang Chen, Yunhe Zhang, and Shiping Wang. Dual fusion-propagation graph neural network for multi-view clustering. *IEEE Trans. Multim.*, 25: 9203–9215, 2023. 4

[23] Jie Xu, Huayi Tang, Yazhou Ren, Liang Peng, Xiaofeng Zhu, and Lifang He. Multi-level feature learning for contrastive multi-view clustering. In *CVPR*, pages 16030–16039, 2022. 4

[24] Jie Xu, Yazhou Ren, Xiaolong Wang, Lei Feng, Zheng Zhang, Gang Niu, and Xiaofeng Zhu. Investigating and mitigating the side effects of noisy views for self-supervised clustering algorithms in practical multi-view scenarios. In *CVPR*, pages 22957–22966, 2024. 5

[25] Weiqing Yan, Yuanyang Zhang, Chenlei Lv, Chang Tang, Guanghui Yue, Liang Liao, and Weisi Lin. Gcfagg: global and cross-view feature aggregation for multi-view clustering. In *CVPR*, pages 19863–19872, 2023. 4

[26] Xiao Yu, Hui Liu, Yan Zhang, Yuxiu Lin, and Caiming Zhang. Hubness-enabled clustering and recovery for large-scale incomplete multi-view data. *ACM Transactions on Knowledge Discovery from Data*, 19(1):1–23, 2024. 3

[27] Kun Zhan, Feiping Nie, Jing Wang, and Yi Yang. Multiview consensus graph clustering. *IEEE Trans. Image Process.*, 28 (3):1261–1270, 2019. 3

[28] Runwu Zhou and Yi-Dong Shen. End-to-end adversarial-attention network for multi-modal clustering. In *CVPR*, pages 14607–14616, 2020. 3