Multi-Modal Proxy Learning Towards Personalized Visual Multiple Clustering

Jiawei Yao[1] Qi Qian[2] Juhua Hu[1]*
[1] School of Engineering and Technology, University of Washington, Tacoma, WA 98402, USA
[2] Alibaba Group, Bellevue, WA 98004, USA
{jwyao, juhuah}@uw.edu, qi.qian@alibaba-inc.com

Abstract

Multiple clustering has gained significant attention in recent years due to its potential to reveal multiple hidden structures of data from different perspectives. The advent of deep multiple clustering techniques has notably advanced the performance by uncovering complex patterns and relationships within large datasets. However, a major challenge arises as users often do not need all the clusterings that algorithms generate, and figuring out the one needed requires a substantial understanding of each clustering result. Traditionally, aligning a user's brief keyword of interest with the corresponding vision components was challenging, but the emergence of multi-modal and large language models (LLMs) has begun to bridge this gap. In response, given unlabeled target visual data, we propose Multi-MaP, a novel method employing a multi-modal proxy learning process. It leverages CLIP encoders to extract coherent text and image embeddings, with GPT-4 integrating users' interests to formulate effective textual contexts. Moreover, reference word constraint and concept-level constraint are designed to learn the optimal text proxy according to the user's interest. Multi-MaP not only adeptly captures a user's interest via a keyword but also facilitates identifying relevant clusterings. Our extensive experiments show that Multi-MaP consistently outperforms state-of-the-art methods in all benchmark multi-clustering vision tasks. Our code is available at https://github.com/Alexander-Yao/Multi-MaP.

1. Introduction

Clustering, which groups data points based on their similarities, has been extensively researched, since huge amount of unlabeled data are becoming more and more available. Traditional methods [2, 16, 19] exploit general-purpose handcrafted features that are not always ideal for specific tasks. Recently, deep clustering algorithms [9, 23, 33] leverage Deep Neural Networks (DNNs) significantly improve performance. However, most algorithms yield a single data partition, while data can exhibit multiple aspects (e.g., color and species as shown in Fig. 1). Traditional multiple clustering algorithms [1, 12] have been developed to generate different partitions for varying applications, demonstrating the ability to identify multiple distinct clusterings from a dataset. Contemporary advancements in the field reveal a growing inclination among researchers to integrate deep learning methodologies for facilitating multiple clustering outcomes. Predominantly, such techniques capitalize on auto-encoders and data augmentation processes to capture a broad spectrum of feature dimensions, thereby enhancing the performance of multiple clustering [18, 27, 35].

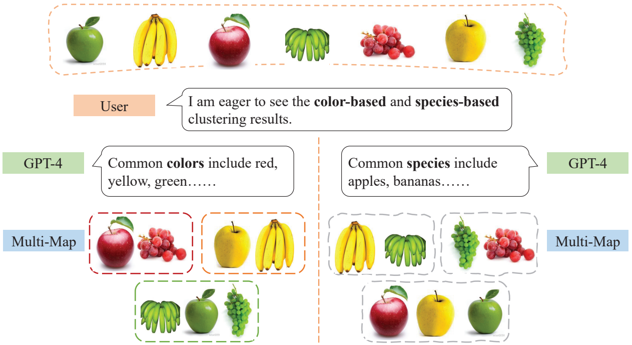

User
I am eager to see the **color-based** and **species-based** clustering results.
GPT-4
Common **colors** include red, yellow, green……
Common **species** include apples, bananas……
GPT-4
Multi-Map
Multi-Map

Figure 1. The flow chart of Multi-MaP. Multi-MaP obtains multiple clustering results based on the high-level concepts from users and the reference words from GPT-4.

However, a common issue arises as users often do not require all the clusterings generated by the algorithm, and identifying the relevant ones necessitates a substantial understanding of each clustering result. Therefore, in this work, we initiate an exploration into a method that is adept at accurately capturing and reflecting a user's interest. Users typically express their interests through concise keywords (e.g., color or species), and aligning these with different visual components precisely is challenging. Fortunately, the advent of multi-modal models like CLIP [25] that aligns images to their corresponding text descriptions, can be helpful to fill this gap. However, unlike methods that employ labeled data to fine-tune pre-trained models [6, 30], multiple

*Corresponding author

clustering deals with environments marked by vague or undefined label categories and amounts. Consequently, given only a high-level concept from the user, it is infeasible to fine-tune the pre-trained models to capture a specific aspect of the data, without the detailed labels corresponding to the user's concept.

An intuitive strategy to integrate pre-trained models into clustering is the zero-shot feature extraction, followed by clustering of the resultant embeddings. However, this approach exhibits limitations, particularly in capturing the interests of users within the dataset. Taking the multi-modal model CLIP [25] as an example, when feeding image data into CLIP, regardless of what aspects of clustering the user expects, CLIP can only produce the same embeddings. Even considering the scenario that different pre-trained models can capture different aspects of the same data as in [9], it is hard to tell which one matches a user's preference. Fortunately, given CLIP's ability to model image-text pairs collaboratively, we can use a user's high-level concept to trigger the corresponding feature extraction from the pre-trained encoders from CLIP. However, no previous work has studied if CLIP has this potential to uncover different aspects of images, which is the focus of this work.

Specifically, we propose to integrate a user's high-level concept describing the preference using a personalized textual prompt. For example, if a user's focus pertains to the color dimension of fruit, a prospective prompt might be formulated as "a fruit with the color of *", wherein the "*" placeholder represents the proxy word awaiting determination using the knowledge in CLIP. Thereafter, we can learn the proxy word embedding by maximizing the similarity between the image and text embedding. However, the proxy word embedding is now searched in a continuous space while the original CLIP used discrete tokens, which can downgrade the performance. We prove that the performance can be well guaranteed by selecting the nearest token as the reference, which is unavailable in a clustering task.

Fortunately, we can use the user's high-level concept as a reference, which however covers a broad range of tokens under its scope. Therefore, we propose to leverage large language models like GPT-4 to generate candidate tokens using the user's high-level concept, in which the closest can be used as a closer reference token. Furthermore, in some scenarios, users may provide multiple concepts to obtain multiple clusterings simultaneously as shown in Fig. 1, we can also introduce a negative loss with these constrastive concepts to further enhance the learning. Therefore, to capture a user's specific interest and discover personalized clustering structure hidden in the data, we propose a multi-modal proxy learning method (Multi-MaP). Multi-MaP incorporates both text prompts and unlabeled images from the clustering task, and leverages CLIP to acquire their respective personalized representations using both reference word and concept-level constraints. The contributions of this work can be summarized as:

- We are the first to explore a deep multiple clustering method that precisely captures a user's interest(s) and generates personalized clustering(s) accordingly.
- We propose a novel multi-modal proxy learning method, Multi-MaP, by leveraging the text and image encoders pre-trained by CLIP, where the user's interest can be captured by the personalized text prompts.
- Considering the challenge of learning a word proxy in a continuous space while tokens in CLIP were discrete, we theoretically prove that a close reference token can help constrain the search, which motivates the proposed reference word constraint and concept-level constraint.
- We conduct extensive experiments on all publicly available visual multiple clustering tasks, which empirically demonstrates the superiority of the proposed Multi-MaP, with a precise capturing of a user's interest.
- Finally, to the best of our knowledge, we are the first who demonstrate that CLIP can uncover different semantic aspects of images.

2. Related Work

In this section, we briefly review two related directions, that is, multiple clustering and multi-modal models.

2.1. Multiple Clustering

Multiple clustering, as a kind of method that can discover alternative perspectives of data, has attracted considerable attention. Traditional methods for multiple clustering [11] use shallow models to find different ways of grouping data. Some of these methods rely on constraints to produce alternative clusterings. For instance, COALA [1] uses the objects in an existing clustering as constraints for creating a new clustering and Qi et al. [21] formulated multiple clustering as a constrained optimization problem. Other methods exploit different feature subspaces to generate multiple clusterings. For example, Hu et al. [12] found multiple clusterings by maximizing the eigengap in different subspaces. MNMF [34] adds the inner product of similarity matrices as a regularization term to find multiple clusterings. Some methods also use information theory to generate multiple clusterings. Gondek [8] applied conditional information bottleneck and Dang [4] used an expectation maximization framework to optimize mutual information.

Recently, some methods have used DNNs to find multiple clusterings and achieved better results. Wei et al. [31] proposed a deep matrix factorization based method that uses multi-view data to find multiple clusterings. ENRC [18] uses an auto-encoder to learn object features and finds multiple clusterings by optimizing a clustering objective function. iMClusts [27] exploits auto-encoders and multi-head attention to learn features from different perspectives, and then find multiple clusterings. AugDMC [35] leverages data augmentation to generate diverse aspects of images and learns the representations to discover multiple clusterings. Although existing deep multiple clustering methods have achieved remarkable results, they require users to exert considerable efforts to select the correct clustering they need. In this work, we aim to efficiently and effectively capture users' interests using short keywords and provide clustering results accordingly.

2.2. Multi-modal Model

Multi-modal learning refers to the process of learning representations from different types of input modalities, such as image data, text, or speech. As related, we focus on how vision models can benefit from natural language supervision. CLIP [25] is a notable model, which is trained with a dataset containing 400 million text-image pairs from the internet. The objective is to align images to their corresponding text using contrastive learning.

Fine-tuning approaches adapt vision-language models like CLIP to specific downstream image recognition tasks. CoOp [36] and CLIP-Adapter [6] exemplify this, with the latter integrating residual style feature blending to enhance performance on various visual classification tasks. Additionally, insights from TeS [30], further elucidate the effectiveness of fine-tuning strategies in leveraging natural language supervision for enhanced visual understanding. Recognizing the scarcity of labeled data for various tasks, significant research efforts have been dedicated to enhancing zero-shot learning. Some approaches extend beyond CLIP by incorporating other large pre-trained models. For instance, VisDesc [17] harnesses the power of GPT-3 to generate comprehensive contextual descriptions corresponding to given class names, thereby demonstrating superior performance compared to CLIP's basic prompts. UPL [13] and TPT [29] leverages unlabeled data to optimize learnable input text prompts. InMaP [24] recovers the proxy of each class in the vision space with the help from the text proxy. All of these methods aim to improve the performance of vision classification tasks, while clustering is a different scenario that we do not have class names that we can exploit to extract useful information from CLIP. In this work, we consider leveraging the CLIP encoder to extract coherent text and image embeddings, and using GPT-4 to integrate users' interests for the purpose of multiple clustering.

3. The Proposed Method

In this section, we first briefly review the training objective in CLIP as follows, and then describe the details of our proposed method based on that.

3.1. Multi-modal Pre-training

Given a set of image-text pairs as $\{x_i, t_i\}_{i=1}^n$, where x_i is an image and t_i is the corresponding text description, their vision and text representations can be obtained by two encoders as $\mathbf{x}_i = f(x_i)$ and $\mathbf{t}_i = h(t_i)$. $f(\cdot)$ and $h(\cdot)$ are vision and text encoders for optimization, where $\mathbf{x}_i$ and $\mathbf{t}_i$ have the unit norm. Then, these two encoders can be learned by minimizing the contrastive loss as

$$\min_{f,h} \sum_i -\log \frac{\exp(\mathbf{x}_i^\top \mathbf{t}_i/\tau)}{\sum_j \exp(\mathbf{x}_i^\top \mathbf{t}_j)/\tau)} - \log \frac{\exp(\mathbf{t}_i^\top \mathbf{x}_i/\tau)}{\sum_j \exp(\mathbf{t}_i^\top \mathbf{x}_j)/\tau)}$$

where τ is the temperature.

This contrastive loss aims to pull the image and its description together while pushing away the irrelevant text [22], which enables the emerging multi-modal applications, e.g., zero-shot transfer [24, 25], text-to-image generation [28], etc.

3.2. Multi-modal Proxy Learning

Given the pre-trained vision and text encoders from CLIP, this work takes one step further to investigate if we can extract user-specific information from the alignment between images and text.

Concretely, given an image of fruit [12] as shown in Fig. 2, some users may be interested in only one specific property of the object, e.g., color. In this scenario, applying the vision encoder to extract the representation for the whole image can miss the preference of users. To mitigate the problem, we propose to explore the proxy representation from the image with the guidance from the text using users' preference, named Multi-Modal Proxy learning (Multi-MaP).

Recall that CLIP is pre-trained by images and text descriptions, where the text prompt is "a photo of a fruit" for an image containing "fruit". Now given a user's preference (e.g., color), we can rewrite the prompt as "fruit with the color of *" denoted by t_i^* for image x_i, where "*" is the proxy word and its token embedding is $\mathbf{w}_i$ that is learnable. Then, we can align image and text prompt representations to obtain the appropriate proxy embedding for a user's interest. Since there is no negative pairs, only the similarity between positive pairs can be optimized as

$$\mathcal{L}(\mathbf{w}_i) = -\langle f(x_i), h(t_i^*)\rangle \tag{1}$$

where the vision and text encoders are frozen, and $\mathbf{w}_i$ is the only variable for learning as the representation of the proxy word. By maximizing the similarity to the image representation, Multi-MaP aims to learn the optimal text proxy according to the user's interest.

However, it should be noted that the text encoder was pre-trained with discrete text tokens, while the domain of $\mathbf{w}_i$ in Eqn. 1 is unconstrained. Therefore, the text representation extracted from the frozen text encoder can be inaccurate for $\mathbf{w}_i$ that degenerates the performance, which is demonstrated in the following theory.

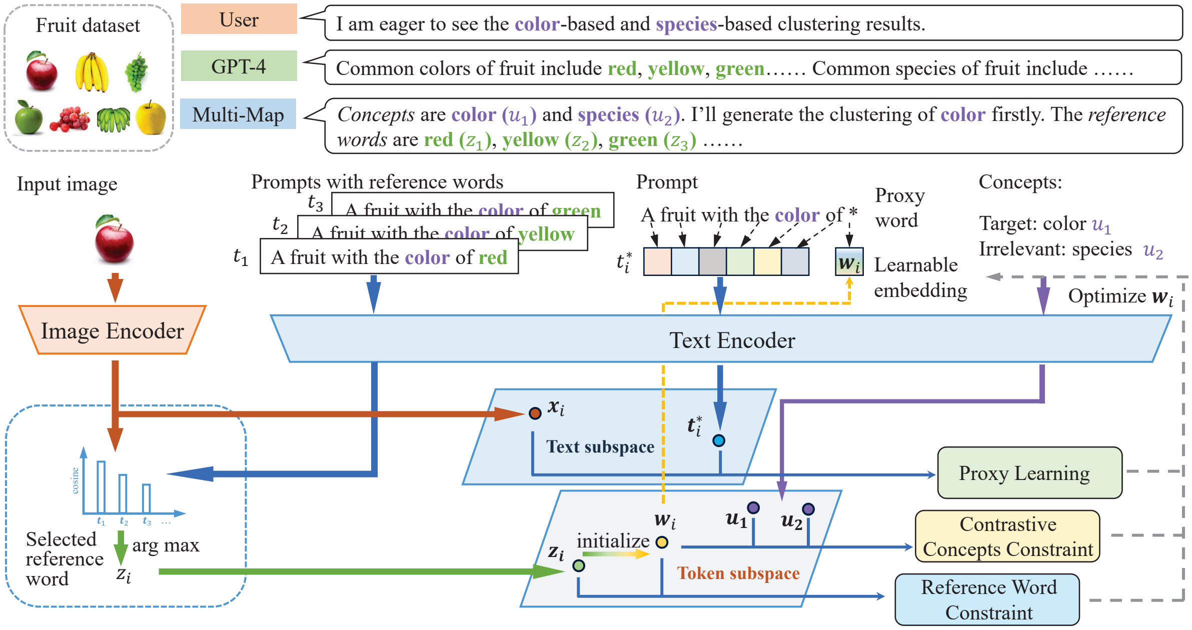

Fruit dataset
User
I am eager to see the color-based and species-based clustering results.
GPT-4
Common colors of fruit include red, yellow, green...... Common species of fruit include ......
Multi-Map
Concepts are color (u_1) and species (u_2). I'll generate the clustering of color firstly. The reference words are red (z_1), yellow (z_2), green (z_3) ......
Input image
Prompts with reference words
t_3 A fruit with the color of green
t_2 A fruit with the color of yellow
t_1 A fruit with the color of red
Prompt
A fruit with the color of *
t_i^*
Proxy word
w_i
Learnable embedding
Concepts:
Target: color u_1
Irrelevant: species u_2
Optimize w_i
Image Encoder
Text Encoder
x_i
t_i^*
Text subspace
cosine
t_1 t_2 t_3 ...
Selected reference word
arg max
z_i
w_i
u_1
u_2
initialize
z_i
Token subspace
Proxy Learning
Contrastive Concepts Constraint
Reference Word Constraint

Figure 2. **Multi-MaP framework.** In the training process of Multi-MaP, the vision and text encoders are frozen and the proxy word embeddings $\mathbf{w}_i$ are learnable. Specifically, it first constructs the prompt embeddings based on the reference words provided by GPT-4 using a user's high-level concept, and then selects a reference word z_i for each image according to the similarity between the prompt embeddings $\mathbf{t}_i$ and the image embeddings $\mathbf{x}_i$. Then, it combines the prompt and the reference words to form the new prompt embeddings $\mathbf{t}_i^*$ and maximizes the similarity to the image representation, so the proxy word embeddings $\mathbf{w}_i$ can capture the desired image features. In addition, the proxy word embeddings $\mathbf{w}_i$ should be close to the target concept word $\mathbf{u_1}$ and the selected reference word $\mathbf{z}_i$ to construct the concept-level constraint and reference word constraint, which capture the features related to the user's interest.

For the sake of simplicity, we assume $h'(t) \in \mathbb{R}$ is defined on the whole set but only has the right estimation on a discrete set as $T = \{t_i\}$ and the counterpart with the unconstrained set is denoted as $H(w)$. According to the definition, we have $\forall t \in T, h'(t) = H(t)$. The gap between the estimation from h' and H on unconstrained variable w can be depicted as follows.

Theorem 1. *Given $w \notin T$ and $t \in T$, if assuming h' and H are L_h and L_H-Lipschitz continuous, we have*

$$\|h'(w) - H(w)\|_2 \leq (L_h + L_H)\|t - w\|_2$$

Proof. According to the definition, we have

$$\begin{aligned}
&\|h'(w) - H(w)\|_2 = \|h'(w) - h'(t) + h'(t) - H(w)\|_2 \\
&\leq \|h'(w) - h'(t)\|_2 + \|h'(t) - H(w)\|_2 \\
&= \|h'(w) - h'(t)\|_2 + \|H(t) - H(w)\|_2 \\
&\leq (L_h + L_H)\|t - w\|_2
\end{aligned}$$

□

Remark Theorem 1 implies that the distance of the estimation $h'(w)$ to the ground-truth result $H(w)$ is bounded by that of w to an arbitrary discrete token t. Therefore, by selecting the nearest token as the reference, the bound can be improved as shown in the following corollary.

Corollary 2. *With the assumptions in Theorem 1 and letting $t' = \arg\min_i \|t_i - w\|_2$, we have*

$$\|h'(w) - H(w)\|_2 \leq (L_h + L_H)\|t' - w\|_2$$

3.2.1 Concept-level Constraint

According to the above analysis, a good reference t' can help guarantee the performance. Fortunately, the input concept (e.g., color) from the user can be leveraged as the reference to constrain the freedom of the proxy word. Therefore, given the target concept word u, we can obtain its token embedding as $\mathbf{u} = \phi(u)$. Then, to learn appropriate representations from the proxy embedding, the original problem can be rewritten with the constraint as

$$\mathcal{L}(\mathbf{w}_i) = -\langle f(x_i), h(t_i^*)\rangle \quad s.t. \quad \|\mathbf{w}_i - \mathbf{u}\|_2^2 \leq \lambda$$

The constrained problem is equivalent to

$$\mathcal{L}(\mathbf{w}_i) = -\langle f(x_i), h(t_i^*)\rangle + \alpha\|\mathbf{w}_i - \mathbf{u}\|_2^2 \tag{2}$$

following [3], which can be optimized effectively by gradient descent.

3.2.2 Constrained Optimization with Reference Word

However, it is well known that the user concept is often with a large scope covering a broad range of words (e.g., color

Datasets	# Samples	# Clusters
ALOI	288	2;2
Card	8,029	13;4
CMUface	640	4;20;2;4
Fruit	105	3;3
Fruit360	4,856	4;4
Stanford Cars	1,200	4;3
Flowers	1,600	4;4

Table 1. Dataset Statistics.

covers all including but not limited to ‘red’, ‘blue’, ‘green’, etc.). As suggested by our above theoretical analysis, it is desired if the reference is as close as possible. In a clustering scenario and given only the user’s high-level concept, it is challenging to find a closer reference word to further constrain the proxy learning. Fortunately, with the development of large language models (LLMs), we can leverage them (e.g., GPT-4) to provide relevant words according to a user’s high-level concept as the candidate set and develop a selection strategy to obtain a closer reference word for each image. While the responses gathered from GPT-4 might not always precisely align with the data’s ground truth, they indisputably furnish valuable candidate features, enriching the capabilities of Multi-MaP.

To elucidate, considering the task of clustering a fruit dataset based on the concept of color, we can pose a query to GPT-4 as “What are the common colors of fruit?”. The response we obtain is “Common colors of fruit include red, yellow, green, orange, purple, and blue”. The colors enumerated in this response can serve as reference word candidates. It is worth noting that while the color spectrum derived from this method might surpass the actual colors present in the data, we collect all of them into the candidate set of reference words $\{z_k\}_k$, where z_1: “red”, z_2: “yellow”, etc. Then, their text representations can be obtained from the prompt t_k as “fruit with the color of z_k”. Given the image x_i, the closest reference can be observed as

$$z_i = \arg\max_k \langle \mathbf{x}_i, \mathbf{t}_k \rangle$$

where $\mathbf{t}_k = h(t_k)$. After that, $\mathbf{w}_i$ can be initialized with the token embedding of z_i as $\mathbf{z}_i = \phi(z_i)$. Moreover, we change the regularization using a closer reference word compared to the high-level concept as

$$\mathcal{L}(\mathbf{w}_i) = -\langle f(x_i), h(t_i^*) \rangle + \alpha \|\mathbf{w}_i - \mathbf{z}_i\|_2^2 \qquad (3)$$

3.2.3 Contrastive Concepts

In some application scenarios, one user may need more than one clustering and provide high-level concepts as $\{u_j\}$, e.g., u_1:“color”, u_2:“species”, etc. For the concept “color”, the irrelevant concept “species” can be leveraged as the negative constraint for the learning of proxy word. Concretely, let u_w denote the target concept word, a contrastive loss can be adopted as regularization

$$R(\mathbf{w}_i) = -\log \frac{\exp(\mathbf{w}_i^\top \mathbf{u}_w)}{\sum_j \exp(\mathbf{w}_i^\top \mathbf{u}_j)}$$

and the final objective becomes

$$\mathcal{L}(\mathbf{w}_i) = \\ -\langle f(x_i), h(t_i^*) \rangle + \alpha \|\mathbf{w}_i - \mathbf{z}_i\|_2^2 - \beta \log \frac{\exp(\mathbf{w}_i^\top \mathbf{u}_w)}{\sum_j \exp(\mathbf{w}_i^\top \mathbf{u}_j)} \qquad (4)$$

where the first term is to infer the user-specific feature, while the latter two terms constrain the proxy word to the reference words for the appropriate representation extraction from the pre-trained text encoder. The overall framework of Multi-MaP is illustrated in Fig. 2.

4. Experiments

To demonstrate our proposed method, we evaluate Multi-MaP on all publicly available image datasets in multiple clustering, including ALOI [7], Stanford Cars [14], Card [35], CMUface [10], Flowers [20], Fruit [12], and Fruit360 [35] as summarized in Table 1.

We compare Multi-MaP against five state-of-the-art methods: **MSC** [12] is a traditional multiple clustering method that uses hand-crafted features; **MCV** [9] leverages multiple feature extractors to represent different “views” of the same data and employs a multi-input neural network to enhance clustering outcomes; **ENRC** [18] is a deep multiple clustering method that integrates auto-encoder and clustering objective to generate different clusterings; **iMClusts** [27] makes use of the expressive representational power of deep autoencoders and multi-head attention to accomplish multiple clusterings; **AugDMC** [35] leverages augmentations to learn different image representations to achieve multiple clustering.

4.1. Experiment Setup

We employ Adam and set momentum as 0.9 to train the model for 1000 epochs. All hyper-parameters are searched according to the loss score of Multi-MaP, where the learning rate is searched in $\{0.1, 0.05, 0.01, 0.005, 0.001, 0.0005\}$, weight decay is in $\{0.0005, 0.0001, 0.00005, 0.00001, 0\}$, α, β are in $\{0.0, 0.1, 0.2, \ldots, 1.0\}$, and λ is fixed as 1 for all the experiments. For the non pre-trained methods, we perform k-means [15] 10 times due to its randomness and evaluate the average clustering performance using two quantitative metrics, that is, Normalized Mutual Information (NMI) [32] and Rand index (RI) [26]. These measures range in $[0, 1]$, and higher scores imply more accurate

Dataset	Clustering	MSC		MCV		ENRC		iMClusts		AugDMC		Multi-MaP	
		NMI	RI	NMI	RI	NMI	RI	NMI	RI	NMI	RI	NMI	RI
ALOI	Color	0.1563	0.3428	0.6982	0.7439	0.9833	0.9892	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Shape	0.2968	0.5199	0.7359	0.8261	0.9732	0.9861	0.9963	0.9989	1.0000	1.0000	1.0000	1.0000
Fruit	Color	0.6886	0.8051	0.6266	0.7685	0.7103	0.8511	0.7351	0.8632	0.8517	0.9108	**0.8619**	**0.9526**
	Species	0.1627	0.6045	0.2733	0.6597	0.3187	0.6536	0.3029	0.6743	0.3546	0.7399	**1.0000**	**1.0000**
Fruit360	Color	0.2544	0.6054	0.3776	0.6791	0.4264	0.6868	0.4097	0.6841	0.4594	0.7392	**0.6239**	**0.8243**
	Species	0.2184	0.5805	0.2985	0.6176	0.4142	0.6984	0.3861	0.6732	0.5139	0.7430	**0.5284**	**0.7582**
Card	Order	0.0807	0.7805	0.0792	0.7128	0.1225	0.7313	0.1144	0.7658	0.1440	0.8267	**0.3653**	**0.8587**
	Suits	0.0497	0.3587	0.0430	0.3638	0.0676	0.3801	0.0716	0.3715	0.0873	0.4228	**0.2734**	**0.7039**
CMUface	Emotion	0.1284	0.6736	0.1433	0.5268	0.1592	0.6630	0.0422	0.5932	0.0161	0.5367	**0.1786**	**0.7105**
	Glass	0.1420	0.5745	0.1201	0.4905	0.1493	0.6209	0.1929	0.5627	0.1039	0.5361	**0.3402**	**0.7068**
	Identity	0.3892	0.7326	0.4637	0.6247	0.5607	0.7635	0.5109	0.8260	0.5875	0.8334	**0.6625**	**0.9496**
	Pose	0.3687	0.6322	0.3254	0.6028	0.2290	0.5029	0.4437	0.6114	0.1320	0.5517	**0.4693**	**0.6624**
Stanford Cars	Color	0.2331	0.6158	0.2103	0.5802	0.2465	0.6779	0.2336	0.6552	0.2736	0.7525	**0.7360**	**0.9193**
	Type	0.1325	0.5336	0.1650	0.5634	0.2063	0.6217	0.1963	0.5643	0.2364	0.7356	**0.6355**	**0.8399**
Flowers	Color	0.2561	0.5965	0.2938	0.5860	0.3329	0.6214	0.3169	0.6127	0.3556	0.6931	**0.6426**	**0.7984**
	Species	0.1326	0.5273	0.1561	0.6065	0.1894	0.6195	0.1887	0.6077	0.1996	0.6227	**0.6013**	**0.8103**

Table 2. Quantitative comparison. The significantly best results with 95% confidence are in bold.

clustering results. The experiments are conducted with GPU NVIDIA GeForce RTX 2080 Ti.

It should also be noted that some data are difficult to obtain corresponding candidate labels from GPT-4 or the labels do not provide semantic features, such as names. For example, for the identity clustering for the CMUface dataset [10], different identities represent different people and the semantic meaning of names should not affect the clustering results. In this case, we randomly extract 10 words from WordNet [5] as reference words, in order to make the candidate labels more distinctive. For instance, we randomly choose "abstain, candid, function, haphazard, knot, luxury, nonchalance, pension, resilience, taciturn" for the above scenario. Furthermore, all publicly available multiple clustering datasets provide each ground-truth clustering a high-level concept, e.g., 'shape', 'pose', etc. Therefore, in the experiment, we directly use them as users' preferences for our evaluation purposes.

4.2. Performance Comparison

In our experiments, after we obtain the proxy word embedding of each image for a desired concept, we feed them to k-means to obtain the corresponding clustering. Since k-means is random, we repeat ten times and the average performance is reported in Table 2. The best results are marked by bold numbers.

We can observe that Multi-MaP outperforms the baselines in all the cases, indicating the superiority of the proposed method. This also shows a strong generalization ability of the pre-trained model by CLIP, which can capture the features of data in different aspects.

Dataset	Clustering	CLIP$_{\text{GPT}}$		CLIP$_{\text{label}}$		Multi-MaP	
		NMI	RI	NMI	RI	NMI	RI
ALOI	Color	0.8581	0.9407	1.0000	1.0000	1.0000	1.0000
	Shape	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Fruit	Color	0.7912	0.9075	**0.8629**	**0.9780**	0.8619	0.9526
	Species	0.9793	0.9919	1.0000	1.0000	1.0000	1.0000
Fruit360	Color	0.5613	0.7305	0.5746	0.7673	**0.6239**	**0.8243**
	Species	0.4370	0.7552	**0.5364**	**0.7631**	0.5284	0.7582
Card	Order	0.3518	0.8458	0.3518	0.8458	**0.3653**	**0.8587**
	Suits	0.2711	0.6123	0.2711	0.6123	**0.2734**	**0.7039**
CMUface	Emotion	0.1576	0.6532	0.1590	0.6619	**0.1786**	**0.7105**
	Glass	0.2905	0.6869	**0.4686**	**0.7505**	0.3402	0.7068
	Identity	0.1998	0.6388	0.2677	0.7545	**0.6625**	**0.9496**
	Pose	0.4088	0.6473	0.4691	0.6409	**0.4693**	**0.6624**
Stanford Cars	Color	0.6539	0.8237	0.6830	0.8642	**0.7360**	**0.9193**
	Type	0.6207	0.7931	**0.6429**	**0.8456**	0.6355	0.8399
Flowers	Color	0.5653	0.7629	0.5828	0.7836	**0.6426**	**0.7984**
	Species	0.5620	0.7553	**0.6019**	0.7996	0.6013	**0.8103**

Table 3. Variants of CLIP. The significantly best results with 95% confidence are in bold.

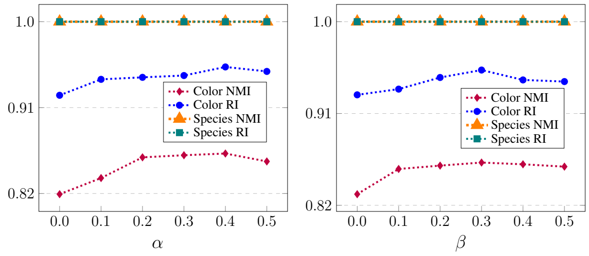
Since our method uses the CLIP encoder and GPT-4 to obtain clustering results, a natural question arises how the performance would be if we directly use them in a zero-short manner. Therefore, we provide two zero-shot variants of CLIP, that is, CLIP$_{\text{GPT}}$ that uses GPT-4 to obtain the candidate labels and predicts labels through zero-shot classification with all candidate labels as class names, and CLIP$_{\text{label}}$ that performs zero-shot classification with all ground truth labels. It should be noted that CLIP$_{\text{label}}$ uses an unfair setting with a ground-truth label set known in advance, which is expected to provide the best performance using CLIP in a zero-short manner. The results are shown in Table 3.

As expected, CLIP$_{\text{label}}$ achieves better performance than CLIP$_{\text{GPT}}$ in almost all cases, since CLIP$_{\text{label}}$ uses a fixed ground truth label set, while CLIP$_{\text{GPT}}$ uses candidate labels that may not match the ground truth exactly, introducing noise. Note that CLIP$_{\text{GPT}}$ and CLIP$_{\text{label}}$ achieve the same results on Cards, because the candidate labels provided by GPT-4 are exactly the same as the true labels.

Besides, Multi-MaP performs better than CLIP$_{\text{GPT}}$ in almost all cases, indicating that the proposed method can learn more effective features through its training process. Moreover, although CLIP$_{\text{label}}$ uses the ground truth, which is expected to be the best, Multi-MaP still outperforms CLIP$_{\text{label}}$ in some cases, such as the clustering of color for

		Multi-MaP$_p$		Multi-MaP$_c$		Multi-MaP$_r$		Multi-MaP$_{cr}$		Multi-MaP	
Modules	Proxy Learning	✓		✓		✓		✓		✓	
	Concept Word	×		✓		×		✓		✓	
	Reference Word	×		×		✓		✓		✓	
	Contrastive Concepts	×		×		×		×		✓	
		NMI↑	RI↑	NMI↑	RI↑	NMI↑	RI↑	NMI↑	RI↑	NMI↑	RI↑
ALOI [7]	Color	0.9619	0.9826	1.0000	1.0000	0.9795	0.9869	1.0000	1.0000	1.0000	1.0000
	Shape	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Fruit [12]	Color	0.7642	0.8439	0.8215	0.9283	0.8136	0.9073	0.8484	0.9308	**0.8619**	**0.9526**
	Species	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Fruit360 [35]	Color	0.5643	0.7665	0.6217	0.7836	0.5910	0.7746	0.6089	0.7965	**0.6239**	**0.8243**
	Species	0.5077	0.7368	0.5137	0.7436	0.5094	0.7425	0.5199	0.7428	**0.5284**	**0.7582**
Card [35]	Order	0.1932	0.8152	0.3568	0.8472	0.3113	0.8229	0.3616	0.8094	**0.3653**	**0.8587**
	Suits	0.2375	0.6282	0.2696	0.6641	0.2498	0.6365	0.2562	0.6599	**0.2734**	**0.7039**
CMUface [10]	Emotion	0.1690	0.6170	0.1714	0.6229	0.1697	0.6360	0.1713	0.6843	**0.1786**	**0.7105**
	Glass	0.3112	0.6911	0.3269	0.7136	0.3162	0.6917	0.3370	0.7108	**0.3402**	**0.7068**
	Identity	0.5617	0.8234	0.6243	0.8359	0.5839	0.8263	0.6391	0.8946	**0.6625**	**0.9496**
	Pose	0.4361	0.6386	0.4550	0.6499	0.4381	0.6429	0.4387	0.6489	**0.4693**	**0.6624**
Stanford cars [14]	Color	0.5939	0.7835	0.6836	0.8659	0.6729	0.8638	0.7112	0.9117	**0.7360**	**0.9193**
	Type	0.5569	0.7996	0.6383	0.8271	0.6091	0.8046	0.6289	0.8181	**0.6355**	**0.8399**
Flowers [20]	Color	0.5783	0.7723	0.5830	0.7833	0.5987	0.7849	0.6216	0.7941	**0.6426**	**0.7984**
	Species	0.5704	0.7608	0.5744	0.7842	0.5723	0.7811	0.5846	0.7892	**0.6013**	**0.8103**

Table 4. Components ablation. All of our components boost performance consistently in all benchmark multi-clustering vision tasks.



(a) Parameter sensitivity of α

(b) Parameter sensitivity of β

Figure 3. Parameter analysis of α and β on Fruit [12].

Fruit360 dataset. This is because CLIP is more inclined to capture the features of one aspect of the data, while Multi-MaP learns better embedding of different aspects by training with the supervision of users' interests. Furthermore, Multi-MaP also achieves very competitive results as CLIP$_{\text{label}}$ in the remaining cases, which further demonstrates the effectiveness of the proposed method.

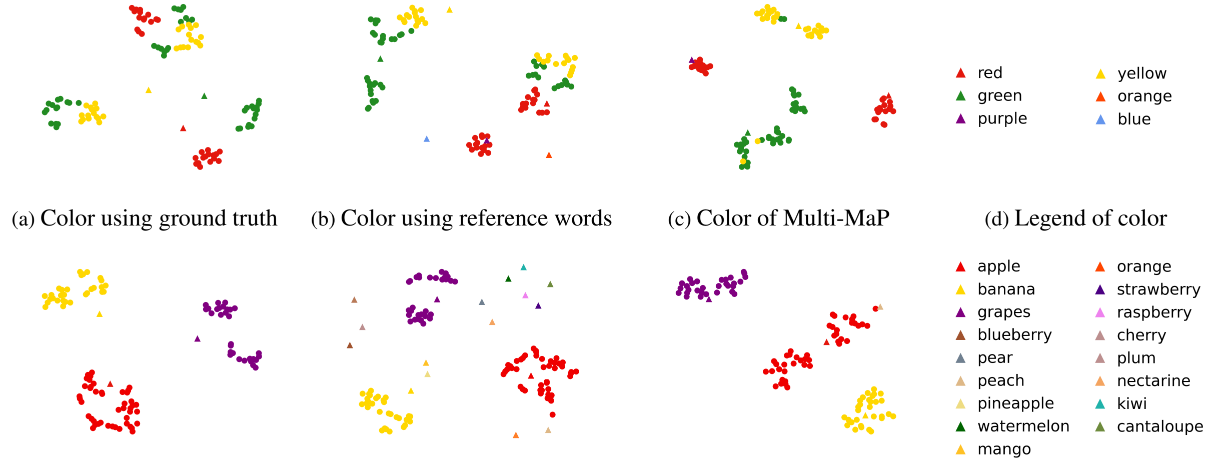
4.3. Ablation Study

To validate the effectiveness of the proposed method, we show the gain from four components in Multi-MaP in Table 4. Let "Multi-MaP$_p$" denote the proxy learning without concept-level constraint, "Multi-MaP$_c$" denote the variant optimized by solely applying concept word to constrain the freedom of the proxy word, "Multi-MaP$_r$" denote the variant optimized by reference word provided by GPT-4 to find a closer reference word to further constrain the proxy learning, and "Multi-MaP$_{cr}$" denote the variant leveraged both concept word and reference word.

We can observe that the proxy learning only with concept word or reference word, i.e., Multi-MaP$_c$ and Multi-MaP$_r$, performs better than Multi-MaP$_p$. This shows that the proposed concept-level constraint and constrained optimization with reference words play an important role in the model as demonstrated by our theoretical analysis. Moreover, the model with combined components, i.e, Multi-MaP$_{cr}$, achieves better results than Multi-MaP$_r$ and Multi-MaP$_c$. This indicates the effectiveness of the combination of reference words and reference concepts. Finally, our proposal using all including the contrastive concepts can further improve the performance, and thus provide the best results on all cases. This further demonstrates our proposal.

4.4. Parameter Analysis

We further investigate the effect of the reference word constraint weight α and concept-level constraint weight β varying from 0.0 to 0.5. The results of Multi-MaP on Fruit datasets are shown in Fig. 3 (a) and Fig. 3 (b), respectively. As α and β change, the proposed method keeps a species score of 1, since the image encoder can capture very effective species features. The above results also show that the proposed method can effectively capture useful information from images, reference words, and target concepts. As α increases, the proposed method first increases and then decreases, and reaches the maximum values at α=0.4. Similar results can be observed for β that the performance of the proposed method first increases and then decreases as β increases, and reaches the maximum values at β=0.3. Indicating a suitable value of α or β is helpful for Multi-MaP to obtain more effective embeddings for multiple clustering tasks. More studies such as efficiency analysis can be found



(a) Color using ground truth (b) Color using reference words (c) Color of Multi-MaP (d) Legend of color

(e) Species using ground truth (f) Species using reference words (g) Species of Multi-MaP (h) Legend of species

Figure 4. Visualization of feature embeddings and related labels. The points represent the image or pseudo-word embeddings, and the triangles represent the prompt or label embeddings. Different colors represent different labels, which are indicated by the text next to the triangles.

in the supplementary.

4.5. Visualization

To further demonstrate the effectiveness of the proposed method, we visualize the representations obtained in CLIP$_{\text{label}}$, CLIP$_{\text{GPT}}$, and Multi-MaP. Specifically, for CLIP$_{\text{label}}$ and CLIP$_{\text{GPT}}$, we visualize the image representations as well as prompts generated with real labels and candidate labels, respectively. For the Multi-MaP, we visualize the word embedding $\boldsymbol{w}_*$ and the candidate labels selected for initialization. The results are shown in Fig. 4. For species clustering, we can see that image embeddings show very clear boundaries and correspond well to the prompt for CLIP$_{\text{label}}$, which indicates that CLIP can effectively capture the features of the species in the data. CLIP$_{\text{GPT}}$ uses the candidate labels to generate prompts, which introduces more noise, but benefits from the CLIP text encoder, the image embedding can keep a relatively far distance from most of the irrelevant prompts. However, since there are still a few images that are labeled as peaches (i.e., a noisy label), it performs slightly worse than CLIP$_{\text{label}}$, as shown in Fig. 4(e). Besides, Multi-MaP can capture the image and users' interests in the training process, therefore it compensates for the shortcomings of CLIP$_{\text{GPT}}$ and achieves better results. On the other hand, for color clustering, the prompts are farther away from CLIP$_{\text{label}}$ and CLIP$_{\text{GPT}}$, that indicates the image embeddings mainly capture the features of species, which have no direct connection with color. CLIP$_{\text{GPT}}$ generates the prompt from the candidate label, which has more noise than the ground truth label, resulting in worse performance than CLIP$_{\text{label}}$. The proposed method can distinguish different colors more clearly, because it can learn from the user's interest and capture the color-related features. However, some red color embeddings are closer to purple, because some images in the datasets are actually purple, but labeled as red. To sum up, the proposed method can learn more effective embeddings based on the users' interests for multiple clustering tasks.

5. Conclusion

To conclude, our study thoroughly investigates the significant challenges that current advanced deep learning techniques face in multiple clustering. A key issue is that users often do not need every clustering result produced by an algorithm, and selecting the most relevant one requires an in-depth understanding of each outcome. To overcome this challenge, the proposed method introduces a novel multi-modal proxy learning process, which effectively aligns a user's brief keyword describing the interest with the corresponding vision components. By integrating a multi-modal model and GPT-4 to precisely capture a user's interest using a keyword, the proposed approach uses both reference word constraint and concept-level constraint to discover personalized clustering result(s), which can also lead to enhanced performance. Experiments on diverse datasets demonstrate the superiority of the proposed method in multiple clustering tasks with a precise capture of a user's interest. The proposed method is limited by data with semantic meaningful labels, although we can use WordNet to help, whose comprehensive study will be our future work.

6. Acknowledgement

This research is supported in part by Advata Gift funding. All opinions, findings, conclusions and recommendations in this paper are those of the author and do not necessarily reflect the views of the funding agencies.

References

[1] Eric Bae and James Bailey. Coala: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In *ICDM*, pages 53–62. IEEE, 2006. 1, 2

[2] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Springer, 2006. 1

[3] Stephen P. Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2014. 4

[4] Xuan Hong Dang and James Bailey. Generation of alternative clusterings using the cami approach. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 118–129. SIAM, 2010. 2

[5] Christiane Fellbaum. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer, 2010. 6

[6] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, pages 1–15, 2023. 1, 3

[7] Jan-Mark Geusebroek, Gertjan J Burghouts, and Arnold WM Smeulders. The amsterdam library of object images. *International Journal of Computer Vision*, 61:103–112, 2005. 5, 7

[8] David Gondek and Thomas Hofmann. Conditional information bottleneck clustering. In *3rd ieee international conference on data mining, workshop on clustering large data sets*, pages 36–42, 2003. 2

[9] Joris Guérin and Byron Boots. Improving image clustering with multiple pretrained cnn feature extractors. In *British Machine Vision Conference 2018, BMVC 2018*, 2018. 1, 2, 5

[10] Stephan Günnemann, Ines Färber, Matthias Rüdiger, and Thomas Seidl. Smvc: semi-supervised multi-view clustering in subspace projections. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 253–262, 2014. 5, 6, 7

[11] Juhua Hu and Jian Pei. Subspace multi-clustering: a review. *Knowledge and information systems*, 56(2):257–284, 2018. 2

[12] Juhua Hu, Qi Qian, Jian Pei, Rong Jin, and Shenghuo Zhu. Finding multiple stable clusterings. *Knowledge and Information Systems*, 51(3):991–1021, 2017. 1, 2, 3, 5, 7

[13] Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022. 3

[14] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013. 5, 7

[15] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982. 5

[16] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, pages 281–297. Oakland, CA, USA, 1967. 1

[17] Sachit Menon and Carl Vondrick. Visual classification via description from large language models. *arXiv preprint arXiv:2210.07183*, 2022. 3

[18] Lukas Miklautz, Dominik Mautz, Muzaffer Can Altinigneli, Christian Böhm, and Claudia Plant. Deep embedded non-redundant clustering. In *Proceedings of the AAAI conference on artificial intelligence*, pages 5174–5181, 2020. 1, 2, 5

[19] Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14, 2001. 1

[20] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008. 5, 7

[21] ZiJie Qi and Ian Davidson. A principled and flexible framework for finding alternative clusterings. In *SIGKDD*, pages 717–726, 2009. 2

[22] Qi Qian, Lei Shang, Baigui Sun, Juhua Hu, Tacoma Tacoma, Hao Li, and Rong Jin. Softtriple loss: Deep metric learning without triplet sampling. In *ICCV*, pages 6449–6457. IEEE, 2019. 3

[23] Qi Qian, Yuanhong Xu, Juhua Hu, Hao Li, and Rong Jin. Unsupervised visual representation learning by online constrained k-means. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16640–16649, 2022. 1

[24] Qi Qian, Yuanhong Xu, and Juhua Hu. Intra-modal proxy learning for zero-shot visual categorization with clip. In *Thirty-seventh Conference on Neural Information Processing Systems, NeurIPS 2023*, 2023. 3

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 2, 3

[26] William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971. 5

[27] Liangrui Ren, Guoxian Yu, Jun Wang, Lei Liu, Carlotta Domeniconi, and Xiangliang Zhang. A diversified attention model for interpretable multiple clusterings. *IEEE Transactions on Knowledge and Data Engineering*, 2022. 1, 2, 5

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10674–10685. IEEE, 2022. 3

[29] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems*, 35:14274–14289, 2022. 3

[30] Junyang Wang, Yuanhong Xu, Juhua Hu, Ming Yan, Jitao Sang, and Qi Qian. Improved visual fine-tuning with natural language supervision. *arXiv preprint arXiv:2304.01489*, 2023. 1, 3

[31] Shaowei Wei, Jun Wang, Guoxian Yu, Carlotta Domeniconi, and Xiangliang Zhang. Multi-view multiple clusterings using deep matrix factorization. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6348–6355, 2020. 2

[32] JV White, Sam Steingold, and CG Fournelle. Performance metrics for group-detection algorithms. *Proceedings of Interface*, 2004, 2004. 5

[33] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487, 2016. 1

[34] Sen Yang and Lijun Zhang. Non-redundant multiple clustering by nonnegative matrix factorization. *Machine Learning*, 106(5):695–712, 2017. 2

[35] Jiawei Yao, Enbei Liu, Maham Rashid, and Juhua Hu. Augdmc: Data augmentation guided deep multiple clustering. In *INNS DLIA@IJCNN*, 2023. 1, 3, 5, 7

[36] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 3